



UNIVERSIDAD DE SEVILLA

ESCUELA TÉCNICA SUPERIOR EN INGENIERÍA DE LA EDIFICACIÓN

DEPARTAMENTO DE INGENIERÍA GRÁFICA  
EXPRESIÓN GRÁFICA ARQUITECTÓNICA

TESIS DOCTORAL:

**OPTIMIZACIÓN EN EL POSICIONAMIENTO PARA LA  
REALIZACIÓN DE UN LEVANTAMIENTO  
ARQUITECTÓNICO**

Elena Cabrera Revuelta

---

Supervisado por:  
José Antonio Barrera Vera  
M<sup>a</sup> José Chávez de Diego





© Joaquín Salvador Lavado, QUINO. Toda Mafalda, Lumen.  
Penguin Random House, España





*A mis padres.*



# Agradecimientos

Por fin ha llegado el momento de dar las gracias, porque aunque esto es lo primero que se lee, es lo último que se escribe.

A las primeras personas que quiero dar las gracias es a mis directores José Antonio y María José. Gracias por confiar en mí y ayudarme tanto durante estos años. Han formado un equipo estupendo y además lo hemos pasado genial, así que muchas gracias por haberme permitido contar con vosotros y por el tiempo dedicado. He aprendido mucho de ambos, no sólo como investigadores sino también como personas.

Por otro lado, han sido muchas las personas que han formado parte de esta investigación. Empecemos por el principio. Gracias a Jorge Urrutia porque fue el primero a que me prestó su ayuda, facilitándome bibliografía por la que empezar con este trabajo. Continúo agradeciendo a Alberto Márquez, quien nos planteó la posibilidad de adentrarnos en el mundo de los Algoritmos Genéticos.

Más adelante conocí a José Ramón Portillo, y a Félix Martín. Quiero dar las gracias a ambos, ha sido un auténtico placer trabajar con ellos en la implementación de uno de los métodos obtenidos.

Y cómo no, tengo que agradecer con mayúsculas al equipo de FORMA-15, tanto a los organizadores de esta iniciativa como al equipo, formado por Lorena, Laura, Juan Carlos, Yago y *Minolo*. Mis agradecimientos son infinitos para los dos últimos, con quienes lo he pasado realmente bien. Me han acogido como una más en la Escuela de Ingeniería Informática. Muchas gracias Yago y *Minolo*, no sé qué habría hecho sin vosotros.

No puedo olvidarme de todas aquellas personas que me han apoyado y que siempre han tenido un rato para escucharme; en especial a mis compañeros del Departamento de Ingeniería Gráfica, sobre todo a Gabi, por sus prácticos consejos y a Marga por sus ánimos y optimismo constantes.

---

También quiero agradecer a todas las personas que me han ayudado tanto del Departamento de Matemática Aplicada I, como del Departamento de Ciencias de la Computación e Inteligencia Artificial, en especial a Fernando Sancho.

A todos mis amigos, que no son pocos, sobre todo a Ana Alicia porque nos hemos apoyado mucho en esto, y a Pablo, por sus ánimos y su ayuda.

Y por último tengo que dar las gracias a mis hermanos Ester, Laura, Quique y Ana, por haberme aguantado estos últimos meses, y a mi hermana Gema en especial, porque se implica en todo como la primera y su opinión es siempre fundamental para mí. A mi tío Antonio por ser un pozo de sabiduría gaditana; a mi prima Leti, que aunque me haya escuchado tanto sigue confundiendo las palabras *algoritmo* y *logaritmo*; a mi primo Manolo por su ayuda; y a mis padres porque todo se lo debo a ellos.



---

# Resumen

La presente Tesis Doctoral desarrolla un método de selección del conjunto de posicionamientos desde los cuales realizar la toma de datos geométricos de un edificio, en aras de ejecutar el Levantamiento Arquitectónico del mismo. Este trabajo de investigación se ha realizado con el fin de obtener un método que asista a los profesionales de la Topografía, la Ingeniería y la Arquitectura en los trabajos de levantamientos de edificios. Para ello, se ha realizado un estudio acerca de las técnicas de optimización adecuadas para conocer el número óptimo de posicionamientos desde los cuales realizar la toma de datos, y la ubicación de los mismos.

Esta investigación se ha creído necesaria al observar que los Levantamientos Arquitectónicos se han venido realizando basándose en la experiencia y buen hacer de los profesionales que lo ejecutan, intuyendo, a ojo, cuáles son los mejores puntos sobre los que estacionar, pero sin ningún control que verifique que la toma de datos vaya a ser correcta. Esto puede conllevar pérdidas económicas, de tiempo, y lo más importante, errores en la toma de datos.

Para dar respuesta al problema planteado, en la presente memoria de tesis se desarrolla el diseño de dos algoritmos, los cuales devuelven un número razonable de posicionamientos que garantizan la visualización de forma correcta de un edificio desde su lado exterior.

El primero de ellos deriva de las técnicas de optimización basadas en el Problema de la Galería de Arte. Dicho problema estudia el número mínimo suficiente de guardias para *vigilar* en su totalidad una sala de un museo. Tras esta primera aproximación, se ha diseñado un Algoritmo Genético, basado en la computación evolutiva e implementado como un Sistema Multiagente.

Una vez diseñados ambos algoritmos, se han probado con éxito en diferentes casos, entre los cuales, por su especial interés patrimonial, se expone como estudio de caso el levantamiento realizado con escáner láser sobre el Molino

---

de Marea del Río Arillo, en Cádiz, declarado Bien de Interés Cultural.

Con este trabajo se ha llegado a la conclusión principal de que los algoritmos diseñados aportan una importante mejora en los trabajos de levantamientos. Por ello, se considera necesario continuar con la línea de investigación iniciada, con el fin de obtener resultados que se ajusten a una tipología más amplia de edificios.

**Palabras Clave:** Optimización, Levantamiento Arquitectónico, Algoritmo Genético, Sistemas Multiagente.



# Abstract

This Phd Thesis sets a selection method for the various positions from which data acquisition is carried out for an Architectural Survey on a building. The aim of this research work was to obtain a method that can be used by Land Surveyors, Engineers and Architects to accurately carry out architectural surveys on buildings. To that end, a method of finding the optimum number of positions from which data acquisition is carried out has been investigated.

This research was deemed necessary when it was observed that up until now, Architectural Surveys have been carried out based on the professional experience and know-how of the person carrying them out, who uses his/her intuition to calculate by sight where the best positions to place the instruments are, but without any control checking that this data acquisition will be accurate. This can lead to a waste of money, time and more importantly, errors in data acquisition.

To solve the problem posed, two algorithms were designed for this research project, to give a reasonable number of positions which allow a building to be accurately visualised from the outside.

The first of these comes from the optimization techniques based on the Art Gallery Problem. This problem studies the minimum number of guards necessary to watch over an art gallery in its entirety. After this first approach, a Genetic Algorithm was designed. This technique, as can be guessed from its name, simulates the Evolution of Species.

Once the algorithms were designed, an architectural survey was carried out with a laser scanner on the Rio Arillo Tidal Mill, in Cádiz, declared a Cultural Heritage Site. This was used as a case study.

The main conclusion obtained from this research is that both of the algorithms designed mean an important improvement to surveying works. Nevert-

---

heless, it is considered necessary to continue this line of research, in order to obtain more accurate results, and for it to be applicable to a wider range of building types.

**Keywords:** Optimization, Architectural Surveying, Genetic Algorithm, Multi Agent Systems.

# Índice general

Resumen . . . . .	VII
Palabras Clave . . . . .	VIII
Abstract . . . . .	IX
Keywords . . . . .	X
<b>Índice</b>	<b>XI</b>
<b>1. Introducción</b>	<b>1</b>
1.1. El Levantamiento Arquitectónico . . . . .	2
1.2. Los Problemas del Proceso del Levantamiento Arquitectónico Tradicional . . . . .	4
1.3. Justificación . . . . .	6
1.4. Tipificación del Problema . . . . .	9
1.5. Estado de la Cuestión . . . . .	12
1.5.1. De los Levantamientos Arquitectónicos . . . . .	12
1.5.2. De las Técnicas e Instrumentos Principales de Levantamiento Tridimensional . . . . .	20
1.5.3. De los Métodos de Optimización en la Ingeniería . . . . .	45
1.5.4. Investigaciones actuales relacionadas con el problema planteado . . . . .	57

<b>2. Objetivos</b>	<b>77</b>
<b>3. Metodología</b>	<b>79</b>
<b>4. Marco Teórico</b>	<b>85</b>
4.1. Revisión bibliográfica sobre los Problemas de Visibilidad . . . . .	86
4.1.1. El Problema de la Galería de Arte . . . . .	87
4.1.2. Variantes del Problema de la Galería de Arte . . . . .	92
4.1.3. Vigilancia en tres dimensiones . . . . .	103
4.1.4. Conclusión . . . . .	104
4.1.5. Aplicación del Problema de la Galería de Arte a los Le- vantamientos Arquitectónicos . . . . .	104
4.2. De los Algoritmos Genéticos (AG) . . . . .	107
4.2.1. Introducción . . . . .	107
4.2.2. Funcionamiento de un Algoritmo Genético . . . . .	108
4.2.3. Diferencias entre los Algoritmos Genéticos y los métodos convencionales de optimización . . . . .	109
4.2.4. Creación de la Población . . . . .	111
4.2.5. Evaluación de la Población . . . . .	113
4.2.6. Operadores Genéticos . . . . .	114
4.2.7. Convergencia . . . . .	119
4.2.8. Algoritmos Genéticos aplicados al Problema de la Gale- ría de Arte . . . . .	120
<b>5. Diseño de Métodos de Optimización en el Posicionamiento para la     Realización de un Levantamiento Arquitectónico</b>	<b>123</b>
5.1. Método Determinista. . . . .	124
5.1.1. Similitudes con el Problema de la Fortaleza . . . . .	124

5.1.2.	Descripción del Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Caso General .	124
5.1.3.	Descripción del Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Polígonos Ortogonales . . . . .	142
5.1.4.	Implementación . . . . .	148
5.1.5.	Resultados . . . . .	150
5.1.6.	Conclusiones . . . . .	152
5.2.	Método basado en Técnicas Metaheurísticas. Primera Aproximación . . . . .	157
5.2.1.	Elementos de los Algoritmos Genéticos . . . . .	159
5.2.2.	Creación de la Población Inicial . . . . .	160
5.2.3.	Codificación . . . . .	161
5.2.4.	Función Objetivo y Valor de Ajuste . . . . .	162
5.2.5.	Selección . . . . .	170
5.2.6.	Cruce . . . . .	171
5.2.7.	Mutación . . . . .	172
5.2.8.	Creación de la Siguiete Generación . . . . .	173
5.2.9.	Creación de Generaciones Sucesivas y Fin . . . . .	174
5.2.10.	Pseudocódigo . . . . .	174
5.3.	Método basado en Técnicas Metaheurísticas. Segunda aproximación e implementación . . . . .	178
5.3.1.	Modelado basado en Agentes . . . . .	178
5.3.2.	Acerca de <i>Net-Logo</i> . . . . .	179
5.3.3.	Diseño del Algoritmo Genético en <i>Net-Logo</i> . . . . .	180
5.3.4.	Funcionamiento del modelo . . . . .	187

5.3.5. Resultados . . . . .	192
5.3.6. Conclusiones . . . . .	200
<b>6. Resultados</b>	<b>203</b>
<b>7. Conclusiones</b>	<b>205</b>
<b>8. Líneas futuras de investigación</b>	<b>209</b>
<b>A. Estudio de caso</b>	<b>211</b>
A.1. Los Molinos de Marea . . . . .	212
A.1.1. Desarrollo histórico . . . . .	212
A.1.2. Funcionamiento . . . . .	214
A.2. El Molino del Río Arillo y su entorno . . . . .	216
A.2.1. El Parque Natural de la Bahía de Cádiz . . . . .	216
A.2.2. Molinos de Marea de la Bahía de Cádiz . . . . .	218
A.2.3. El Molino de Marea del Río Arillo . . . . .	220
A.3. Ejecución del levantamiento arquitectónico . . . . .	229
A.3.1. Trabajos previos . . . . .	229
A.3.2. Toma de Datos . . . . .	230
A.3.3. Post-procesado y resultados obtenidos . . . . .	232
A.3.4. Conclusión . . . . .	233
<b>B. Estudio de ángulos</b>	<b>235</b>
B.1. Toma de datos . . . . .	237
B.2. Manejo de nubes de puntos . . . . .	238
B.3. Error calculado y análisis de los datos . . . . .	241

## ÍNDICE GENERAL

---

C. Protocolo ODD	243
D. Código del modelo en <i>Net-Logo</i>	251
Lista de figuras	279
Referencias	291





# Capítulo 1

## Introducción

El presente trabajo es una *investigación de frontera*, dado que se utilizan metodologías y conceptos de las **Matemáticas Aplicadas** atípicos en la **Expresión Gráfica Arquitectónica**.

Dentro de la Expresión Gráfica Arquitectónica, la investigación se ubica en el área de los **Sistemas de Representación Técnicos y la Topografía**, y en concreto, trata sobre la mejora en los procesos de trabajo en los levantamientos arquitectónicos.

Es sabido que las Matemáticas están divididas en distintas áreas, entre las que se encuentran las **Ciencias de la Computación**, la **Inteligencia Artificial** o la **Geometría Computacional**. El término *Matemática Aplicada* se refiere a aquellos métodos y herramientas matemáticas que pueden ser utilizados en el análisis o resolución de problemas pertenecientes al área de las ciencias básicas o aplicadas.

Uno de los campos que más interés ha suscitado en la comunidad científica relativo a la Geometría Computacional, es el **Problema de la Visibilidad**, o también conocido como *Problema de Vigilancia*. Este problema incluye todos los conceptos de iluminación y vigilancia de estructuras geométricas, en todas sus posibles variantes y nacen con el denominado **Problema de la Galería de Arte** (Danaraj and Klee, 1974).

Por otro lado, se han estudiado métodos de optimización basados en la **Computación Evolutiva**, y en concreto, los **Algoritmos Genéticos**, realizando un modelado a través de **Sistemas Multiagentes**.

## 1.1. Punto de Partida: El Levantamiento Arquitectónico

La **Arqueología** es la ciencia que estudia el ser humano y su legado, tales como edificios, herramientas y el arte. Es una ciencia histórica que suscita un alto interés social. Prueba de ello es la posibilidad de encontrar yacimientos arqueológicos en todo el mundo, los cuales sirven para contar la historia de la humanidad que aconteció en diferentes lugares y momentos (Borrmann et al., 2015).

Con el fin de preservar nuestro legado, en la actualidad existe una creciente conciencia sobre la conservación y documentación de bienes arquitectónicos y arqueológicos. Pues bien, la operación a través de la cual se realiza la toma de datos geométricos de un bien de interés para su estudio, documentación y análisis, se conoce como **Levantamiento**.

La obra de Antonio Almagro denominada *Levantamiento Arquitectónico*, (Almagro Gorbea, 2004) puede considerarse como una de las más utilizadas como referencia en el Área de los Levantamientos Arquitectónicos, propiamente dicho.

En ella, se define este término como «...el conjunto de operaciones, de medidas y de análisis necesarios para comprender y documentar el bien arquitectónico en su configuración completa».

Continuando con el análisis del concepto que se realiza en la obra de Antonio Almagro, cabe destacar que un levantamiento arquitectónico debe permitir, entre otras cosas:

- Un conocimiento preciso y fiable de la configuración morfológica y dimensional del objeto.
- Un conocimiento técnico, tecnológico y material del objeto, con el cual se comprendan sus modalidades constructivas y las alteraciones y modificaciones que ha sufrido.
- Una ágil edición de la planimetría del edificio.

Para llevar a cabo la realización de un levantamiento arquitectónico, los profesionales tienen a su alcance diferentes instrumentos y formas de trabajo: métodos tradicionales, estaciones totales, cámaras fotográficas y hasta los

## 1.1. EL LEVANTAMIENTO ARQUITECTÓNICO

---

más novedosos escáneres láser. La posibilidad de combinar diferentes instrumentos y técnicas permite que sea alcanzable conseguir representaciones de los edificios u objetos arqueológicos con un nivel de detalle máximo (Véase Figura 1.1).

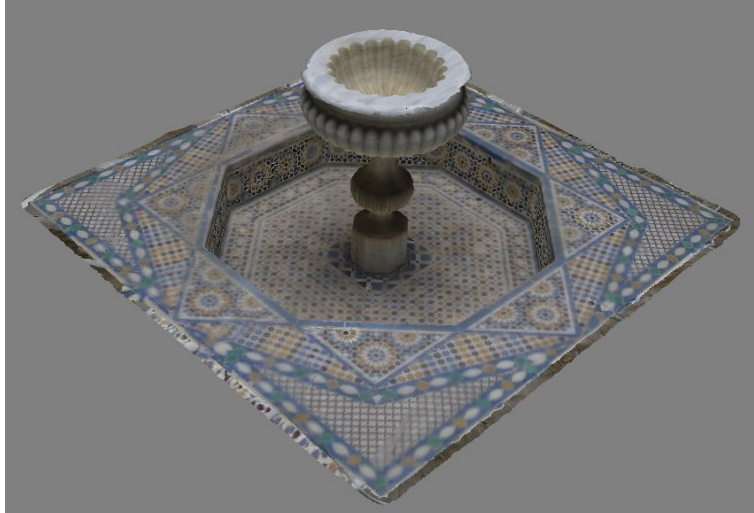


Figura 1.1: Fuente Pabellón de Marruecos de 1929, Sevilla. Modelo obtenido a través de técnicas de Fotogrametría (Ornia Núñez et al., 2017).

La **Topografía** se define como el arte de describir y delinear detalladamente la superficie de un terreno (Real Academia Española, 2014). Hasta hace relativamente poco, para llevar a cabo trabajos de topografía se han venido utilizando métodos e instrumentos que podrían clasificarse como artesanales (Ríos, 2000), como la cinta métrica, la plomada, el nivel de agua, o la tirantez.

El avance en la tecnología de los instrumentos topográficos, así como en los programas informáticos que permiten la visualización de modelos tridimensionales a partir de fotos y a partir de nubes de puntos, ha permitido que la representación sea más detallada y rápida. Pero, no se debe perder de vista que la fiabilidad de un levantamiento no sólo depende de lo novedosos que sean los instrumentos que se utilicen, sino también de quién los maneja y de saber combinar correctamente las diferentes técnicas.

## 1.2. Los Problemas del Proceso del Levantamiento Arquitectónico Tradicional

Haciendo nuestro el estudio realizado por el profesor José Antonio Barrera Vera en su tesis *Aplicación de tecnologías innovadoras en la documentación geométrica del Patrimonio Arquitectónico y Arqueológico* (Barrera Vera, 2006) donde aborda cuestiones en el contexto de los levantamientos arquitectónicos, centrándose en el proceso de los aspectos geométrico-dimensionales de la documentación del Patrimonio Histórico, se puede decir que las diferentes fases de las que se compone el proceso de levantamiento tradicional son las siguientes:

1. **Trabajo de campo.** En esta fase se realiza la toma de datos *in situ* de los datos geométricos necesarios para posteriormente concebir la reproducción del edificio o del objeto a levantar.
2. **Trabajo de Gabinete.** Tras la toma de datos, es necesario post-procesarlos para lograr definir el modelo geométrico del objeto que se pretende levantar.
3. **Presentación.** Se elabora la presentación gráfica del objeto de la forma más adecuada, en cuanto a la escala, los colores, el formato, etc.

Sin embargo, en ninguna de estas fases se considera la planificación previa que debiera tener el proceso de levantamiento.

En la fase de Trabajo de Campo, la recopilación de datos para realizar un Levantamiento Arquitectónico es un trabajo tedioso. Esto incluye encontrar la mejor posición para realizar la toma de datos, mover el equipo de trabajo, etc. Este procedimiento conlleva a un consumo de tiempo que se repite para cada posicionamiento (Borrmann et al., 2015).

Al carecer de una buena planificación, en la toma de datos es más que probable que se emplee más tiempo del que fuera necesario; o que no se tomen todos los datos de forma correcta; incluso puede ocurrir que se tomen datos redundantes. Esto daría como resultado la obtención de datos tridimensionales de mala calidad, los cuales no permitirán un post-procesado satisfactorio.

En el caso de emplear un escáner láser para la toma de datos, el problema es añadido: al tratarse de un sistema de captura de datos masivos, los archivos obtenidos son extremadamente densos y pesados, de forma que una toma de

datos eficiente es más que necesaria. Esta cuestión es abordada en el artículo «A Randomized Art-Gallery Algorithm for Sensor Placement» (González-Baños, 2001), en el cual se plantea que se debe tener en cuenta que la adquisición de datos tridimensionales de alta calidad es una operación complicada, y su post-procesado puede ser costoso computacionalmente, dado que las nubes de puntos obtenidas incorporan una gran cantidad de datos. Esto lleva a que se plantee la siguiente pregunta: ¿Cómo deberían ser colocados los sensores utilizados (cámaras, escáner láser, estación total) para conseguir la información necesaria para construir un modelo visual y geométrico, de la forma más eficiente posible?

Se concluye pues que una correcta planificación de los posicionamientos desde los cuales realizar la toma de datos puede influir de forma positiva en los siguientes aspectos:

- Reducción del tiempo de realización del trabajo y, a su vez, del coste del mismo.
- Mejora de la calidad de los datos obtenidos.
- Reducción de datos redundantes.
- Seguridad de que la toma de datos ha sido completa.
- Post-procesado más rápido y eficiente.

### 1.3. Justificación

El tema a desarrollar en el presente trabajo de investigación queda justificado a través de tres ideas fundamentales.

La primera de ellas consiste en que en la actualidad la creación de modelos tridimensionales parecidos a la realidad, así como su visualización, se está convirtiendo en algo cada vez más común (Alsadik et al., 2012). La realidad aumentada, las visualizaciones virtuales de la realidad (Véase Figura 1.2) o las impresiones 3D (Véase Figura 1.3), son elementos que cada día son más demandados, y abogan por tener una fuerte presencia en los próximos años.



Figura 1.2: Modelo tridimensional de la fortaleza de Al Zubarah, Qatar, generada con el *software* Agisoft PhotoScan (Kersten et al., 2015).



Figura 1.3: Capitel impreso en 3D. Elaboración Propia.

El vertiginoso desarrollo de la tecnología computacional y los cada vez mejores dispositivos de adquisición de imágenes, han hecho que las visualizaciones tridimensionales adquieran más popularidad en diversas áreas, como pueden ser: imágenes médicas, imágenes satelitales o diseño asistido por or-

denador, entre otros.

Cabe destacar el crecimiento de los programas de bajo coste, o incluso *software* libre que existen en el mercado, los cuales permiten generar reconstrucciones automáticas de un objeto, utilizando tan solo una cámara de fotos.

Por otro lado, la segunda idea trata de que «Todo levantamiento debe ser proyectado previamente, su reelaboración debe ser directa, y sus resultados deben estar sujetos a comprobación.» (Almagro Gorbea, 2004). Se cree que es necesario establecer un protocolo de actuación que sea aplicable a cualquier edificio que se desee levantar, con el fin de planificar el trabajo. Dicho protocolo debe estar al alcance de cualquier profesional, garantizando que el resultado del levantamiento cumplirá con las expectativas deseadas. Para ello, partiendo del conocimiento del correcto funcionamiento de los instrumentos que se van a emplear, se plantea la posibilidad de establecer unos pasos a seguir para una correcta planificación.

Y, por último, también debe tenerse en cuenta la tercera idea: la creciente demanda en el campo de la Ingeniería por bajar los costes de producción, ha llevado a los ingenieros a buscar métodos rigurosos para la toma de decisiones, haciendo uso de técnicas de optimización, para diseñar y crear productos y sistemas de manera económica y eficiente. Por lo que las técnicas de optimización, que han alcanzado un alto grado de madurez en los últimos años, están siendo utilizadas en una amplia gama de industrias (Rao, 2009).

Continuando con el análisis que hace el profesor Rao, se debe tener en cuenta el hecho de que en los últimos años, se han desarrollado nuevos métodos de optimización que son conceptualmente diferentes a las técnicas tradicionales de programación matemática. Estos métodos han sido etiquetados como *Métodos Modernos* o *No Tradicionales* de Optimización. La mayoría de ellos se basan en ciertas características y comportamientos de los agentes biológicos, moleculares, enjambres de insectos, o sistemas neurobiológicos (Véase Figura 1.4).

Entre estas Técnicas de Optimización *No Tradicionales*, destacan los **Algoritmos Genéticos**, basados en los principios de la Evolución de la Especie y de la Selección Natural. En los últimos años son numerosas las investigaciones que incorporan esta técnica de optimización, suscitando un gran interés entre los desarrolladores de programas y la comunidad científica.

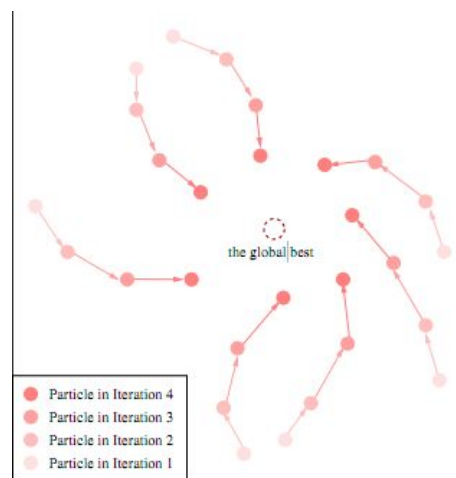


Figura 1.4: Optimización por enjambre de partículas (Sancho Caparrini, 2015).



### 1.4. Tipificación del Problema

El problema que se plantea en el presente trabajo de investigación es el de optimizar el número y la ubicación de los posicionamientos desde los cuales realizar la toma de datos completa de un edificio, para la ejecución del levantamiento arquitectónico del mismo, utilizando instrumentación topográfica para ello. Este problema se encuadra dentro de la Topografía, y en cuanto a su tipificación bajo el punto de vista matemático, forma parte de los denominados **Problemas de Vigilancia**, también conocidos como *Problemas de Iluminación* o *Problemas de la Galería de Arte*.

Los **Problemas de Vigilancia** constituyen el campo de la Geometría Computacional que engloba todos los problemas que de alguna manera están relacionados con la iluminación o vigilancia de cualquier estructura o elemento geométrico.

Claramente el problema de optimizar el número de posicionamientos se enmarca en el área conocida como Optimización Combinatoria, la cual aborda problemas de optimización en los que el conjunto de posibles soluciones es discreto o se puede reducir a un conjunto discreto. El objetivo de dichos problemas consiste en encontrar la mejor solución posible existente, o una solución que minimice o maximice una función objetivo dada. A medida que la complejidad del problema aumenta, debido a la naturaleza del mismo, el coste de ejecución de los algoritmos creados para su resolución puede aumentar, llegando a convertirse en inviables. Este tipo de problemas son *intratables*, bajo el punto de vista computacional.

Para afrontar este tipo de problemas y ante la imposibilidad de una resolución viable, se buscan soluciones aproximadas o subóptimas, pero alcanzables en un tiempo razonable. Dentro de las técnicas de aproximación se encuentran los **Algoritmos Heurísticos**, los cuales proporcionan soluciones de buena calidad para problemas combinatorios complejos, explotando el conocimiento del dominio de aplicación. Por otra parte, los **Algoritmos Metaheurísticos** son estrategias inteligentes generales para el diseño o mejora de los procedimientos heurísticos, para la resolución de problemas con un alto rendimiento.

Siempre que se trabaja con algoritmos, es imprescindible identificar la **Complejidad Computacional** del mismo.

Cuando en computación se habla de complejidad, no se está haciendo referencia a lo difícil que es diseñar un algoritmo que dé solución a un problema. La complejidad está relacionada con dos medidas: el tiempo y el espacio. La

complejidad computacional de un problema es una medida de los recursos computacionales necesarios.

La complejidad de un problema depende de si ha sido posible diseñar un algoritmo con complejidad polinomial, o no, para dar solución al mismo. El matemático canadiense Jack Edmonds, en 1965, introdujo la idea de que aquellos problemas que son computables en tiempo polinomial, son problemas *tratables*, denominando a los algoritmos de tiempo polinomial *buenos algoritmos*. A este tipo de problemas se les designó con la letra  $P$ .

Posteriormente, conforme los medios para computar estos problemas han ido avanzando, se han definido diferentes clases de problemas:

- $P \rightarrow$  Los problemas que se resuelven en tiempo polinomial por una máquina determinista. Esta clase corresponde a la idea intuitiva de los problemas que se pueden resolver de modo efectivo en los casos más complejos.
- $NP \rightarrow$  Los problemas que se resuelven en tiempo polinomial por una máquina no determinista. La principal propiedad de los problemas de esta clase es que sus posibles soluciones pueden ser comprobadas de modo eficiente en tiempo polinomial.

Realmente, la clase de problemas  $P$  está dentro de la clase de problemas  $NP$ , puesto que, si para un problema existe un algoritmo determinista polinomial que lo resuelva, también debe existir un algoritmo no determinista polinomial que lo resuelva. Por lo que un problema  $P$  es también un problema  $NP$ .

Además, dentro de la clase  $NP$ , existe un conjunto de problemas, equivalentes entre sí, respecto a su complejidad computacional, que en la actualidad son intratables y se les denomina  $NP$ -completo (Problemas de Decisión). Para ninguno de ellos se conocen algoritmos polinomiales de resolución, por lo que se les aplica métodos aproximados. Cada problema  $NP$ -completo tiene asociado un problema  $NP$ -duro (Problemas de Búsqueda) con la misma intratabilidad computacional. En la Figura 1.5 puede verse un esquema de los tipos de problemas en función a su complejidad.

Lo que aún no se ha podido demostrar es que  $P$  y  $NP$  coincidan. Por eso, se considera que la clase  $P$  es una subclase de  $NP$ . Si se demuestra que un problema es estrictamente  $NP$ , es decir, que no pertenece a  $P$ , se deberán invertir todos los esfuerzos en diseñar algoritmos heurísticos para conseguir una buena solución, la cual será subóptima, pero alcanzable.

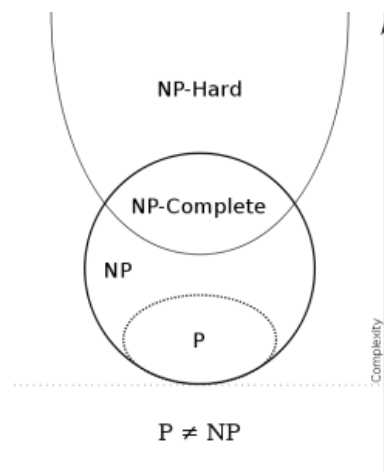


Figura 1.5: Complejidad computacional de los problemas (Sancho Caparrini, 2009).

El problema que se plantea en esta tesis es el de vigilar el borde de un polígono desde el lado exterior del mismo. Los **Problemas de Vigilancia de Polígonos** suelen ser problemas de complejidad computacional *NP*-duro, por lo que se abren dos líneas de investigación: el desarrollo de algoritmos que establecen soluciones aproximadas, las llamadas heurísticas, y la determinación de soluciones óptimas para clases especiales de polígonos simples (Bajuelos Domínguez et al., 2008).

## 1.5. Estado de la Cuestión

En esta sección se abordan las siguientes cuestiones: los Levantamientos Arquitectónicos (Apartado 1.5.1); las Técnicas e Instrumentos Principales de Levantamiento Tridimensional (Apartado 1.5.2); los Métodos de Optimización que se aplican en la Ingeniería (Apartado 1.5.3); y, por último, un análisis de diferentes investigaciones que se están desarrollando en los últimos años relacionadas con el tema que se presenta (Apartado 1.5.4).

### 1.5.1. De los Levantamientos Arquitectónicos

En este apartado, en primer lugar se introduce el concepto de *levantamiento*; más adelante se continúa con un análisis del proceso de trabajo para la ejecución de trabajos de levantamiento; por último, se realiza una breve descripción acerca de los medios de adquisición de datos geométricos que se emplean.

#### 1.5.1.1. Del Concepto de Levantamiento

En primer lugar, en esta sección se trata de definir qué se entiende por el término **Levantamiento Arquitectónico**.

De las definiciones que se pueden encontrar en el *Diccionario de la Real Academia Española* (Real Academia Española, 2014) de la palabra *levantamiento*, la que puede referirse al tipo de levantamiento que se aborda en esta investigación, es la primera.:

*m. Acción y efecto de levantar o levantarse.*

De entre las distintas acepciones del verbo *levantar*, la que más se adecua al concepto que se quiere estudiar es la que sigue:

*9. tr. Proceder a dibujar un plano de una población, una construcción, etc., según procedimientos técnicos.* (Real Academia Española, 2014).

En el *Diccionario de la Real Academia Española* no aparece recogido el significado que para los profesionales de la Topografía, la Ingeniería y la Arquitectura tiene esta palabra. Se encuentra por tanto, una carencia de un significado amplio de este término. Este hecho conlleva al desconocimiento de esta labor

por parte de las personas ajenas a la misma.

Resulta necesaria una definición más completa, la cual sirva para especificar cuestiones como la proporcionalidad, el rigor, la exactitud, la documentación archivística e histórica; en definitiva, el uso y el fin que conlleva realmente un levantamiento en su totalidad.

Para poder completar la definición del término, sería necesario incluir que esta actividad «...se vale de todas las ciencias y todas las técnicas que puedan contribuir a la lectura, a la medida y al análisis de la arquitectura en sus aspectos morfológicos, materiales, y estructurales, patentes u ocultos» (Almagro Gorbea, 2004, p.50).

### Carta del Rilievo.

Con el fin de crear una definición más completa para este término, en 1998 se formó un grupo de trabajo por iniciativa del profesor Cesare Cundari, de la Universidad de la Sapienza de Roma. Dicho grupo estaba formado por expertos procedentes de España, Italia y Francia. Ellos fueron Antonio Almagro, Gianni Carbonara, Stella Casiello, Secondino Coppo, Cesare Cundari, Gaspare de Fiore, Mario Docci, Mario Fondelli, Tatiana Kirova, Emma Mandelli, Paola Rota Rossi Doria, Ciro Robotti, Jean Paul Saint-Aubin y Francesco Starace.

El resultado del trabajo de este grupo de expertos fue el documento conocido como la *Carta del Rilievo*. El documento fue aprobado en el Congreso *Il rilievo dei beni architectonici per la conservazione*, celebrado en Nápoles en abril de 1999.

Se puede encontrar una traducción reducida de dicho documento, elaborada por la arquitecta Ana Almagro Vidal, en la obra *Levantamiento y análisis de edificios: tradición y futuro* (Martín and Puerto, 2003).

Tomando como referencia la obra mencionada anteriormente, se define el concepto de *Levantamiento Arquitectónico* como «...la forma primigenia de conocimiento y por lo tanto el conjunto de operaciones, de medidas y de análisis necesarios para comprender y documentar el bien arquitectónico en su configuración completa...».

Continuando con el análisis recogido en el documento al que se hace referencia, se encuentra una definición más extensa, que define el término como «...el conjunto de investigaciones y operaciones orientadas a determinar las características significativas –bajo los aspectos morfológico, dimensional, figurativo y tecnológico- de un organismo edificado o de un conjunto urbano, a

evaluarlo y a investigarlo, con el propósito de construir un modelo tridimensional simplificado, a través del cual se pueda analizar la obra, facilitando así la interpretación de sus fases de transformación y de los diversos aspectos referidos a los temas más representativos. El levantamiento es, por tanto, un proceso que debe llevar al conocimiento profundo de la obra en estudio, con el fin de poner en evidencia todos sus valores, tanto geométricos como dimensionales, figurativos como estructurales, desde los materiales empleados y las técnicas constructivas, hasta las condiciones de degradación y las relaciones con el contexto urbano».

La realización de levantamientos arquitectónicos completos es una importante labor, a través de la cual se pretende garantizar la Memoria Histórica y la correcta conservación del Patrimonio Cultural (Véase Figura 1.6).



Figura 1.6: Modelo tridimensional de fragmento de escultura de mármol perteneciente al Museo Arqueológico de Paestum (Italia). Elaboración Propia.

### **Levantamiento Gráfico.**

Como se ha puesto de manifiesto, el concepto de levantamiento engloba un conjunto de actividades diversas, tales como la investigación documental, los métodos y técnicas utilizadas, las modificaciones que ha sufrido el edificio, y por supuesto el análisis geométrico. De todas estas actividades, las que más afectan al presente trabajo de investigación son la relacionadas con la elaboración de la documentación gráfica, así como con los medios y las técnicas de adquisición de datos para elaborarla.

La finalidad del **Levantamiento Gráfico** no es más que la de obtener una representación del objeto arquitectónico en forma de modelo a escala reducida, a partir de la cual obtener la documentación gráfica necesaria para su análisis y estudio (Véase Figura 1.7).

Esta representación, tradicionalmente se ha llevado a cabo mediante el



Figura 1.7: Maqueta de Miguel Ángel para la cúpula de San Pedro del Vaticano (Almagro Gorbea, 2004, p.35).

dibujo como elemento primordial, sirviendo éste como soporte para realizar mediciones. Hoy en día, la tecnología existente ofrece diferentes posibilidades, como la creación de modelos tridimensionales con información asociada (Véase Figura 1.8), e incluso vídeos animados a través de los cuales se puede realizar un recorrido virtual e interactivo. Se deben entender estas formas de representación novedosas como complementarias a los planos, pero nunca como técnicas que suplan a las tradicionales. En estas representaciones del bien arquitectónico ha de quedar constancia del estado actual y real del mismo, con las deformaciones y cambios que el edificio haya sufrido a lo largo del tiempo.

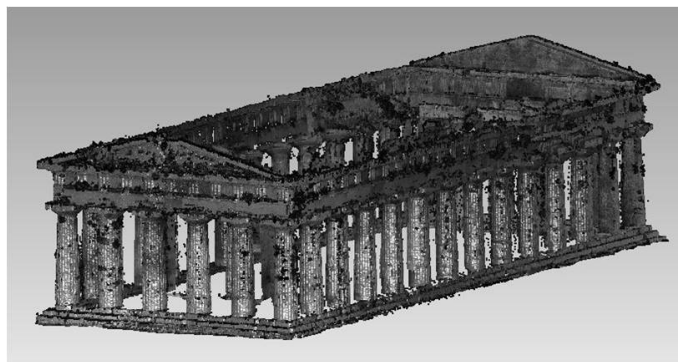


Figura 1.8: Nube de puntos del Segundo Templo de Hera, Paestum (Italia). Elaboración Propia.

Además, cabe mencionar el creciente desarrollo de la filosofía BIM-*Building Information Modeling* y GIS-*Geographic Information System*: «En los últimos años tanto la filosofía BIM como la GIS han adquirido una gran importancia. Ello se debe a la opción de asociar tantas bases de datos como se desee a datos espaciales, dando la posibilidad de crear un modelo transparente en el que se relacionen todos los datos necesarios para comprender un bien arquitectónico en su configuración completa. De esa forma, es posible aglutinar en un único modelo BIM o GIS toda la información necesaria sobre un Bien Patrimonial» (Camillo Saviello et al., 2017).

#### 1.5.1.2. Del Proceso

Como se ha mencionado anteriormente, el proceso por el que se lleva a cabo la realización del modelo simplificado del edificio, consta de un trabajo de campo para la toma de datos geométricos; posteriormente, un trabajo de gabinete, en el cual se procesan los datos recogidos; y, la fase final, la de representación, en la cual se obtiene el resultado final: la documentación gráfica. Posteriormente, ese trabajo es incorporado en algún proyecto multidisciplinar acompañándose de estudios históricos que sitúen al trabajo en el contexto adecuado.

Tal y como aparece reflejado en la obra *Levantamiento y análisis de edificios: tradición y futuro* (Martín and Puerto, 2003, p.65) es usual que «...los datos se capturan mediante operaciones bastante tradicionales y con instrumentación básicamente manual,(...) Se toma un croquis, confiando en la vista del croquizador para recoger el atributo figura y una cierta aproximación al de tamaño, se establecen suposiciones acerca de las horizontales y verticales, se acotan las aristas, los huecos y las diagonales, y se mide con cinta u otro procedimiento más moderno».

Aunque la tecnología actual permite que la toma de datos sea mucho más exacta y ágil que el proceso descrito en el párrafo anterior, llama la atención que en un trabajo técnico el procedimiento esté basado en «la vista del croquizador», dicho de otro modo, en la experiencia de los profesionales, careciendo de técnicas que los auxilien. En la misma línea, puede traducirse del artículo «Network Design» (Fraser, 1996), que el correcto diseño de una red de posicionamientos es alcanzada, gracias, en gran parte, a la experiencia y al sentido común de las personas que realizan este tipo de trabajos.

#### **Problema Detectado.**



Un levantamiento debe ser fiable en cuanto a sus medidas y su representación. Por ello, deberá ser proyectado, su realización será directa, y los resultados verificados.

Resulta necesaria, por el hecho de optimizar los recursos de los que se dispone, una buena planificación del trabajo de campo. De este modo será posible estimar el tiempo que se va a invertir en la realización de la toma de datos, los recursos necesarios y las técnicas que son más adecuadas. Los equipos que se utilizan para la realización de este tipo de trabajos suelen ser de coste elevado, y es muy común el arrendamiento de los mismos, o la subcontratación de empresas especializadas en manejar estos aparatos, así como el *software* apropiado. Por lo que una mala planificación de la toma de datos repercute negativamente en el coste económico del trabajo.

El diseño de una buena planificación deriva en una correcta gestión del tiempo de trabajo y, por tanto, económica, además de contar con la seguridad de que el resultado gráfico será totalmente satisfactorio. Se podrá garantizar el paso a la siguiente fase, la de Gabinete, con la certeza de haber tomado todos los datos necesarios. Además, la eficiencia de los datos obtenidos ayudará a que el post-procesado sea más ágil y se produzcan menos errores.

Para una correcta planificación de la toma de datos, hay que tener en cuenta que el entorno en el que está situado el edificio influye en la misma. El terreno sobre el que esté situado el edificio puede ser irregular o estar en pendiente. Además, los obstáculos que haya a su alrededor pueden ser variables (árboles, otros edificios o zonas intransitables) de forma que la ejecución del levantamiento no solo dependerá del edificio en sí, sino también de los múltiples factores cambiantes que, literalmente, lo rodean.

### 1.5.1.3. De los Medios de Adquisición

Para realizar un levantamiento arquitectónico es necesario pensar en qué medios de adquisición son los más adecuados para el trabajo a realizar. Normalmente se combinan diferentes medios en un solo proceso, siendo habitual utilizar desde métodos tradicionales, como la plomada, hasta las últimas tecnologías, como los escáneres láser, en un mismo trabajo.

Los diferentes medios de adquisición se pueden clasificar en:

- **Medios Tradicionales.** La ventaja que presentan es la necesidad de contacto directo con el objeto, lo que posibilita un conocimiento en profundi-

dad del mismo. Por otro lado, su uso tiene serias limitaciones, son lentos y laboriosos para alcanzar resultados de calidad (Véase Figura 1.9).



Figura 1.9: Instrumentos simples de medición (Almagro Gorbea, 2004, p.42).

- **Medios Topográficos.** El contacto es indirecto. La principal ventaja de estos medios es la exactitud y precisión con la que se conocen las coordenadas de los puntos leídos. Como inconveniente destaca que la toma de datos no es de forma masiva y que se precisa de equipos profesionales, de alto coste y de difícil manejo (Véase Figura 1.10).



Figura 1.10: Estación Topográfica de última generación. Fuente: Leica Geosystem.

- **Medios Fotogramétricos.** La Fotogrametría Terrestre o Fotogrametría de Objeto Cercano es la técnica más vinculada a los levantamientos en materia de Patrimonio en el presente y pasado reciente (Barrera Vera, 2006). La principal ventaja es el fácil acceso a los equipos necesarios:

basta con tener acceso a una cámara de fotos convencional. Además, la toma de datos se realiza de forma masiva. Sin embargo, los resultados no son tan precisos como con otro tipo de instrumentos, y depende en gran medida de la calidad de la imagen. Se necesita un *software* especializado para poder procesar los datos (Véase Figura 1.11).



Figura 1.11: Trabajo de Fotogrametría realizado con el *software* Agisoft Photoscan (Ramírez Pérez and Cabrera Revuelta, 2016).

- **Otros Medios: Nuevos Medios de Adquisición.** Principalmente se trata de los escáneres láser (Véase Figura 1.12).

La diferencia con las estaciones topográficas radica en que la adquisición de datos es masiva y con una alta resolución. Además incorpora la información del color y la reflectividad del material capturado. La gran resolución de puntos obtenidos, la rapidez de la toma de datos, y el *software* que incluyen estos aparatos, permiten la visualización de los resultados en tiempo real, de forma muy realista. Además, su rango de trabajo es amplio, llegando hasta los 350 metros. El inconveniente de estos medios de adquisición es el elevado coste y la utilización de *software* muy específico y complejo, así como la necesidad de profesionales especializados en el manejo de estos datos.

Otra clasificación posible de los Medios de Adquisición es la que hace José Antonio Barrera en su tesis doctoral (Barrera Vera, 2006):



Figura 1.12: Escáner Láser de última generación. Fuente: Leica Geosystem.

- **Medios directos.** Aquellos que permiten realizar *in situ* un croquis susceptible de servir de fuente suficiente para la realización del plano deseado por métodos gráficos. La operativa apropiada es la trilateración y los instrumentos a emplear son los tradicionales: la cinta métrica, con el complemento de la plomada y el nivel. No obstante, existen instrumentos actualmente que los sustituyen, como son los niveles láser y los distanciómetros electrónicos manuales.
- **Medios indirectos.** Con ellos no se obtienen *in situ* dimensiones que permitan la construcción gráfica directa. Son los sistemas que realizan toma de datos masiva de puntos de la realidad, obteniendo desde situaciones auxiliares, la información de la posición espacial relativa entre el punto de estación y el punto observado. Son los teodolitos, los taquímetros, las estaciones totales, escáneres tridimensionales y técnicas fotogramétricas. A partir de las posiciones relativas se obtienen, posteriormente en gabinete, las coordenadas espaciales de cada punto a levantar.

Una vez conocidas las técnicas existentes, es necesario conocer qué herramientas se van a utilizar, teniendo en cuenta que en función a ello el coste y la precisión del trabajo ha de variar.

### 1.5.2. De las Técnicas e Instrumentos Principales de Levantamiento Tridimensional

Dentro de los distintos medios de adquisición y técnicas existentes para llevar a cabo un levantamiento arquitectónico, esta investigación se centra en aquellos con los que se pueden obtener datos de forma masiva, y

capaces de generar nubes de puntos a partir de las cuales crear modelos tridimensionales del objeto.

Como se ha comentado anteriormente, la creación de modelos tridimensionales parecidos a la realidad, así como su visualización, se está convirtiendo hoy en día en algo cada vez más común, en la era digital en la que vivimos; especialmente en la documentación del Patrimonio Histórico, la Restauración, la Conservación y el Desarrollo de la investigación en este campo (Alsadik et al., 2012).

Los sensores tridimensionales son herramientas que permiten generar una imagen tridimensional de la escena, calculando las coordenadas X,Y,Z de cada punto detectado en un corto periodo de tiempo. Algunos sistemas permiten asociar también la información del color de los puntos. Los modelos creados ofrecen un fuerte poder de representación de la realidad, bastante fiables.

Estos modelos tridimensionales son imprescindibles para los estudios de conservación y difusión de bienes culturales. Permiten analizar la deformación de los elementos, el alineamiento de fragmentos de restos arqueológicos, la fabricación de una réplica, la reconstrucción de restos arqueológicos e incluso simular el efecto de un fenómeno específico, como una iluminación artificial o el comportamiento estático por el cambio de algún elemento de la estructura. Es posible calcular el centro de gravedad de un objeto. También permite la integración de estos modelos en entornos virtuales de realidad aumentada o de realidad virtual.

Esta sección se centra en la adquisición de datos a través de sensores tridimensionales basados en técnicas de Fotogrametría y de Escáneres Láser.

### 1.5.2.1. La Fotogrametría

«La Fotogrametría es el arte, la ciencia y la tecnología, que obtiene información real sobre objetos físicos y el medio ambiente a través de procesos de grabación, medida e interpretación de imágenes fotográficas y patrones de energía radiante electromagnética y otros fenómenos» (Slama et al., 1980).

La **Fotogrametría** es una técnica de Triangulación Óptica, la cual permite obtener información geométrica y de color del objeto.

Cuando un punto del espacio tridimensional  $P$  es fotografiado desde dos o más puntos de vista distintos, dicho punto  $P$  es proyectado en las distintas imágenes en una posición distinta. Suponiendo que desde el centro

de la perspectiva parte un rayo que pasa por el punto  $P$  y por la proyección de dicho punto sobre el plano de proyección, es posible reconstruir dichos rayos de visión. El lugar donde esos rayos de visión se cortan es el lugar en el espacio donde está situado el punto  $P$  (Véase Figura 1.13).

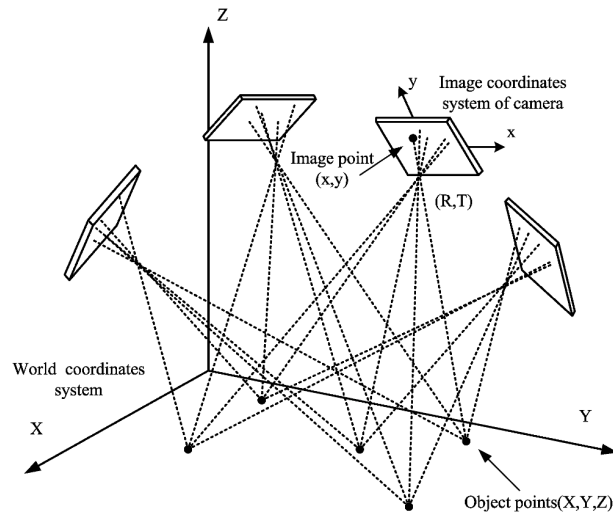


Figura 1.13: Reconstrucción de rayos de visión para obtención del punto en el espacio (Tang et al., 2012).

Cada una de las cámaras está orientada de distinta forma en el espacio. Por otro lado, el objeto estará orientado según su propio sistema de coordenadas cartesianas, el cual puede coincidir con algún sistema de coordenadas predefinido.

La Fotogrametría Estereoscópica imita al comportamiento de la visión binocular, o también llamada visión estereoscópica. Este tipo de visión la poseen aquellos animales que tienen sus órbitas oculares en la parte frontal de la cabeza, como los humanos o los primates.

El cerebro recibe una doble imagen de cada objeto, cada una proveniente de un ojo, como se muestra en la Figura 1.14. El cerebro tiene la capacidad de fundirlas en una sola imagen y es capaz de percibir la profundidad de los elementos.

Esto ocurre debido a que entre los ojos existe una distancia determinada. Por lo tanto, una misma escena es visualizada desde dos ángulos ligeramente distintos. A este efecto de desviación angular de la posición aparente de un objeto, dependiendo del punto de vista elegido se le conoce como **Paralaje**.

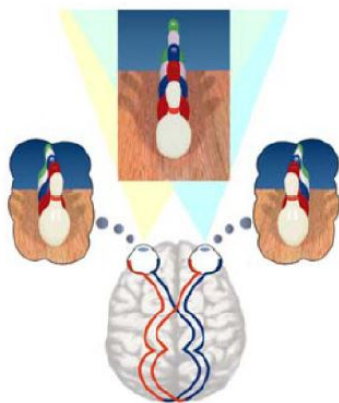


Figura 1.14: Interpretación del cerebro debido a la superposición de imágenes tomadas desde posiciones ligeramente desviadas (Tucci et al., 2013).

La visión monocular, es decir, con un solo ojo, no es suficiente para apreciar la tridimensional.

La Fotogrametría Estereoscópica, por tanto, precisa de la toma de pares de imágenes desde dos puntos de vista ligeramente distintos. La distancia entre pares de fotografías recibe el nombre de *Base-B*, y la distancia desde el par de fotos al objeto fotografiado es denominada *Altura-H*. Para conseguir una adecuada restitución estereoscópica la relación  $B/H$  debe estar bien definida, siendo de entre  $1/3$  a  $1/20$ .

Esta tecnología ha aumentado el rendimiento y automatización de muchos de los procedimientos de restitución, de forma que se ha vuelto una técnica básica para la Arqueología y la Arquitectura. En las últimas décadas, la fotogrametría se ha consolidado como un método con suficiente rigor para la tecnología de medidas tridimensionales (Russo et al., 2011).

### **Ventajas de la Fotogrametría.**

Esta técnica presenta la ventaja fundamental de que es posible medir objetos sin necesidad de tocarlos, pudiendo así medir objetos tóxicos, móviles o que se encuentran a temperaturas extremas. Además, cabe destacar que para ciertos trabajos es suficiente contar con una cámara de fotos convencional para llevar a cabo la toma de datos geométricos.

Se trata de una técnica barata, que ofrece resultados realistas, flexible, rápida y con un alto grado de detalle.

### **Aplicaciones de la Fotogrametría.**

La fotogrametría es utilizada en muchos campos, desde aplicaciones industriales tradicionales a los videojuegos, las producciones de cine, documentación del Patrimonio Cultural, la Medicina o la Investigación Criminal.

A través de la Fotogrametría es posible obtener, entre otros, los siguientes elementos:

- Modelos digitales del terreno (MDT).
- Ortofotos.
- Modelos tridimensionales.
- Cartografía.
- Medidas industriales.
- Animaciones.
- Réplicas.

### División de la Fotogrametría.

En la obra *Network Design and Optimization in Close Range Photogrammetry* (Marshall, 1989), se hace una extensa introducción al término de Fotogrametría, sus aplicaciones, así como los diferentes tipos que existen.

Las imágenes pueden tomarse desde el suelo, o bien desde aviones, globos<sup>1</sup>, cometas, helicópteros, vehículos aéreos no tripulados o incluso desde satélites.

En función a la distancia desde la cámara al objeto fotografiado, la fotogrametría se divide en dos tipos:

- **Aérea.** La cámara está a una distancia mayor a 300 metros del objeto a reconstruir. Sirven para producción de cartografía. Asociada a la Fotogrametría Estereoscópica.
- **De Objeto Cercano o Terrestre.** La cámara está a una distancia menor a 300 metros del objeto a reconstruir. Asociada a la Fotogrametría Monoscópica.

### Principios Fundamentales de la Fotogrametría Terrestre.

---

<sup>1</sup>La primera fotografía aérea fue realizada desde un globo, por el fotógrafo francés Gaspard-Félix Tournachon (1820-1910), más conocido como Nadar.



## 1.5. ESTADO DE LA CUESTIÓN

En fotografía, la imagen formada en la foto es una función de la proyección perspectiva de puntos desde el objeto que se fotografía a la película fotográfica.

Haciendo referencia a la Figura 1.15, una proyección perspectiva es aquella en la cual todos los puntos son proyectados en un Plano de Referencia (*Reference Plane*) a través de un único punto, el cual es llamado Centro de la Perspectiva (*Perspective Centre*). El plano de referencia es la película fotográfica, y el centro de la perspectiva es el objetivo, el cual, en una cámara de fotos se encuentra en la lente.

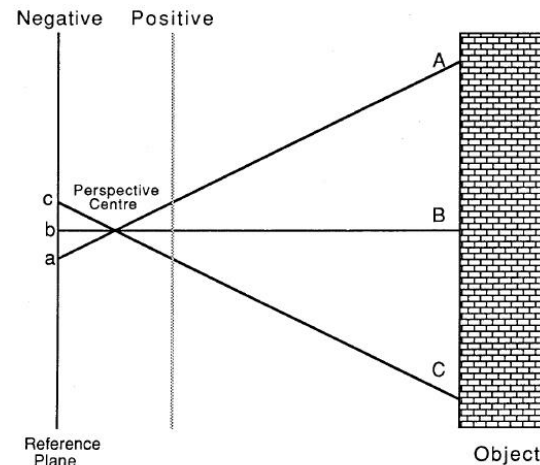


Figura 1.15: Esquema de Proyección Perspectiva (Marshall, 1989).

La orientación de la cámara se determina por dos conjuntos de parámetros.

- Parámetros interiores de orientación:
  - La Distancia Focal,  $d$ , es decir, la distancia existente entre el plano de referencia y el punto de vista.
  - Las coordenadas del punto principal,  $x_0, y_0$ . Este punto es la proyección del punto de vista sobre el plano de proyección.
  - Los parámetros de distorsión geométrica de la lente,  $K_1, K_2, K_3, P_1$  y  $P_2$ <sup>2</sup>.
- Parámetros exteriores de orientación.

Se trata de la localización geográfica de una cámara y la dirección de sus ejes. Los parámetros que definen la posición de la cámara en

<sup>2</sup>Estos parámetros se calculan mediante la calibración de la cámara.

el espacio son  $X_L$ ,  $Y_L$  y  $Z_L$ . La orientación de los ejes viene dada por tres ángulos de rotación, con respecto al sistema de coordenadas del objeto a restituir,  $\omega$  (rotación con respecto al eje X),  $\varphi$  (rotación con respecto al eje Y) y  $\kappa$  (rotación con respecto al eje Z).

### Condición de Colinealidad.

La relación matemática existente entre el objeto y las coordenadas proyectadas en la imagen de ese mismo objeto, se establece haciendo uso de la relación de **colinealidad**.

Esta condición declara que, en el instante de la toma fotográfica, el punto del centro de proyección  $P[X_0, Y_0, Z_0]$ , el punto proyectado en la imagen  $P[x, y]$  y el punto del objeto  $P[X, Y, Z]$ , son colineales, es decir, forman una línea recta, como se muestra en la Figura 1.16.

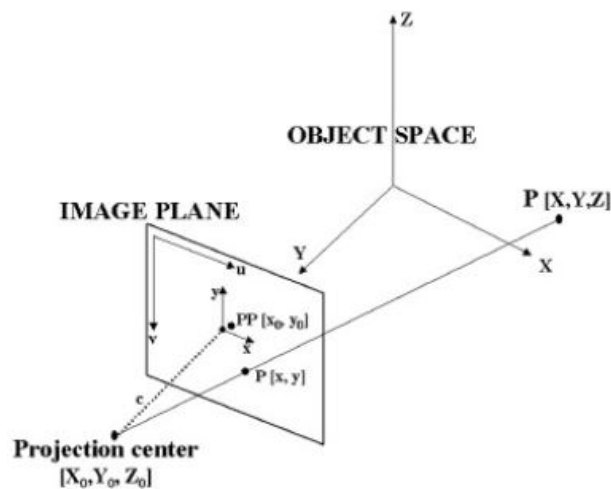


Figura 1.16: Principio de la Condición de Colinealidad (Russo et al., 2011).

Esta condición es fundamental durante el proceso de reconstrucción fotogramétrica, y por lo tanto, se utiliza para formular la relación de transformación entre el objeto y las coordenadas de la imagen. Por supuesto, todos los puntos en la imagen son coplanarios.

De esta condición nacen las **Ecuaciones de colinealidad**, las cuales, son el fundamento de la Fotogrametría. Obligan a que el centro de proyección, el punto proyectado en la imagen y el punto real fotografiado se encuentren alineados en el instante en el que se produce la toma fotográfica. De esta forma, partiendo de dos imágenes de un mismo punto, haciendo que se cumpla esta condición, queda asegurada la intersección de dos rayos homólogos en el punto del terreno prefijado.

**Ajuste simultáneo - *Bundle Adjustment*.**

El **Ajuste simultáneo**, conocido como el *Bundle Adjustment* fue desarrollado por Brown en 1958 (Brown, 1958). Es el proceso que se utiliza para la reconstrucción tridimensional de un objeto a partir de fotografías. En él, se tienen en cuenta todas las imágenes de forma simultánea, reconstruyendo el rayo de visión de cada cámara en el espacio. Por supuesto, está basado en las Ecuaciones de Colinealidad. De esta forma, se localiza la ubicación de la cámara en el espacio, y se localiza, por tanto, el objeto a reconstruir. En la Figura 1.17 se pueden apreciar los siguientes elementos: 5 puntos del objeto original  $\{P_1, \dots, P_5\}$ ; 3 cámaras, cuyos centros de perspectivas son los puntos  $O_1, O_2$  y  $O_3$ ; y, por último, también se pueden apreciar las proyecciones de los puntos del objeto original sobre cada uno de los planos de proyección de las cámaras fotográficas  $\{p_1, \dots, p_5\}$ .

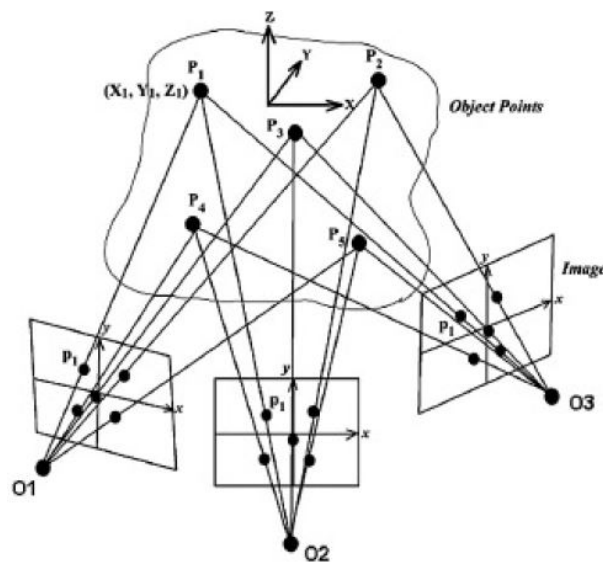


Figura 1.17: Principio de la triangulación fotogramétrica o *Bundle Adjustment* (Russo et al., 2011).

El *Bundle Adjustment* ofrece la precisión más alta de las que se consigue con cualquier otro método de ajuste fotogramétrico (Marshall, 1989). Esto es debido a:

- La posibilidad de corregir errores sistemáticos en la imagen fotográfica.

- La posibilidad de recuperar con gran precisión la orientación interna de cada fotografía.
- La reconstrucción precisa del modelo en el espacio debido a la mayor redundancia en la solución.
- La capacidad de manejar más de dos fotografías.
- La capacidad de manejar fotografías convergentes.
- La integridad de la solución matemática.

#### 1.5.2.2. Escáner Láser

Un **Escáner Láser 3D** puede ser definido como un instrumento capaz de adquirir las coordenadas tridimensionales de los puntos de una porción de superficie de un objeto distribuidas regularmente, de forma automática y con alta densidad (Nobile, 2012). Estos instrumentos son idóneos cuando se trata de obtener información geométrica sobre una superficie de forma compleja.

La información geométrica obtenida por un escáner láser se calcula de la misma que forma que una estación total tradicional, asociando estas tres medidas a un punto leído:

- El ángulo del azimut, que se refiere al ángulo con respecto al eje Y del sistema de coordenadas definido.
- El ángulo cenital, que se refiere al ángulo con respecto al eje Z.
- La distancia.

Sin embargo, el escáner es además capaz de medir la reflectividad del material. La mayoría de ellos ofrece la posibilidad de asociar a cada punto leído el color del mismo, a partir de fotografías tomadas con cámaras que incorpora el propio escáner, para dar la textura real a la nube de puntos obtenida. Las cámaras que incorporan los escáneres láser no suelen ser de gran calidad, por lo que es frecuente el uso de cámaras panorámicas que completen el trabajo realizado por el escáner.

Estos aparatos se pueden clasificar de la siguiente forma, atendiendo al principio en el que basan la medida de la distancia.

- Por Tiempo de Vuelo (*Time of Flight*).  
Este tipo de escáner basa su funcionamiento en la medición del tiempo que emplea el rayo láser en viajar y en volver. La velocidad

de la luz es conocida, así que conociendo el tiempo del viaje de ida y vuelta, se determina la distancia recorrida por la luz. Esta distancia se ha de dividir entre dos. La precisión de un escáner láser de tiempo de vuelo depende de la exactitud con la que es capaz de medir tiempos infinitesimales.

La distancia al punto  $d$ , es calculada conociendo el tiempo de vuelo  $t$  y la velocidad de la luz  $C$ .

$$d = C * \Delta t / 2$$

- Por Diferencia de Fase.

Mide la diferencia de fase entre la luz emitida y la recibida, y utiliza dicha medida para estimar la distancia al objeto. El escáner emite una onda modulada. Este tipo de escáner tiene una velocidad de adquisición de datos superior, por este motivo se suele utilizar para aplicaciones dinámicas (sobre plataformas en movimiento: trenes, automóviles, etc).

- Por Triangulación.

En este tipo de escáner existen tres elementos: el emisor del rayo láser, el sensor y el punto láser sobre la superficie a escanear, formando un triángulo. Como aparece reflejado en la Figura 1.18, la distancia entre el sensor (lente) y el emisor del láser es fija y es conocida como  $b$ . El ángulo del vértice del emisor de láser también es conocido como  $\alpha$ . El ángulo del vértice del sensor,  $\beta$  puede ser determinado al conocer la ubicación del punto del láser reflejado en él, conocida la distancia focal  $f$  y la posición  $p$  del punto proyectado. Estos tres valores permiten determinar el resto de las dimensiones del triángulo y, por tanto, la posición de cada punto en el espacio.

Este sistema es de una precisión muy elevada para determinados ángulos. El alcance de estos escáneres es muy corto, entre 20 y 30 centímetros.

La precisión de la medida de la distancia está relacionada con la distancia del aparato al objeto escaneado, y con la resolución del escaneo. Se establece una resolución determinada para una distancia concreta. Si la distancia a la que se encuentra el objeto es menor, se obtendrá un escaneo con una mayor resolución a la establecida; en cambio, si la distancia a la que se encuentra el objeto es mayor a la que se ha establecido para la resolución, los puntos leídos estarán a una distancia entre ellos mayor a la resolución establecida, como puede verse en la Figura 1.19.

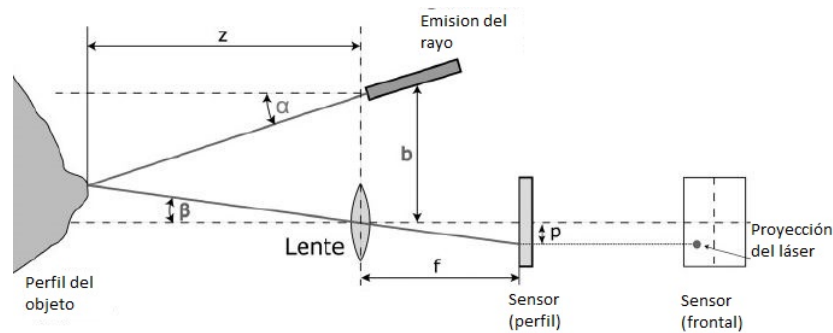


Figura 1.18: Esquema de funcionamiento de un escáner láser de triangulación (Russo et al., 2011).

Por otro lado, se ha de tener en cuenta que el escaneo será más preciso cuanto más frontal sea el objeto con respecto al escáner. De otra forma, el rayo láser incidirá sobre la superficie de forma oblicua, como puede observarse en la Figura 1.20.

Normalmente, para poder escanear un edificio completo, se necesita más de un escaneo. Es frecuente utilizar dianas de materiales con gran reflectividad, las cuales son escaneadas con una precisión mayor que el escaneo general. Las dianas se distribuyen no alineadas entre ellas, formando entre ellas un triángulo lo más isósceles posible, y visibles desde dos posicionamientos distintos. De esta forma, no sólo se obtienen puntos de control con mayor precisión, sino que además permite la unión de distintos escaneos de una forma más sencilla. El proceso de unión de distintas nubes de puntos es conocido como **registro**.

El registro de los distintos escaneos se realiza utilizando el *software* específico que facilite el fabricante del escáner. El registro de nubes de puntos se puede hacer con dianas, o por puntos homólogos, así como por superposición de nubes de puntos con un buen solape entre ellas.

Una vez registradas todas las nubes de puntos, se han de post-procesar los datos geométricos para obtener documentación gráfica. Una de las desventajas principales de esta técnica es que de la nube de puntos no se pueden extraer automáticamente las aristas que definen al objeto. También se debe tener en cuenta que el manejo de la gran cantidad de datos que se genera no es una tarea sencilla.

Por otro lado, se ha de mencionar el último aparato que está revolucionando la toma de datos masiva. Se trata del **ZEB**<sup>3</sup>. Su uso no se encuentra

<sup>3</sup>Desarrollado por la empresa australiana *GeoSLAM*

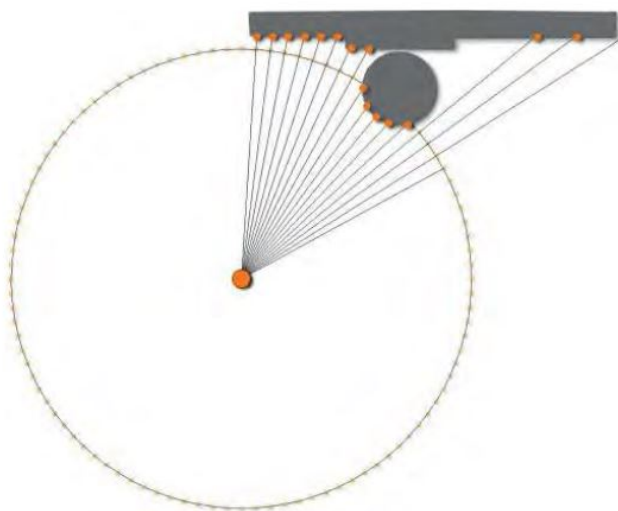


Figura 1.19: Resolución de un escaneo (Nobile, 2012).

muy extendido aún, aunque ya existen dos modelos en el mercado: El ZEB1 (Véase Figura 1.21), y el ZEB-REVO (Véase Figura 1.22). Se caracteriza por sus reducidas dimensiones y su escaso peso, menor a 1 kg. Está pensado para portarlo de forma manual, pero también puede ser acoplado a drones, pértigas y cualquier otro elemento. Es capaz de leer más de 40000 puntos por segundo. La desventaja principal que tiene es su corto alcance. Es óptimo para galerías subterráneas donde un escáner láser no puede ser estacionado. Sin embargo, para trabajos exteriores se necesita hacer un gran recorrido para obtener todos los datos. Es necesario un *software* específico para el post-procesado de los datos (Itidi, 2017).

### 1.5.2.3. Ejemplos de aplicación recientes

- Pasajes subterráneos de la Primera Guerra Mundial. Trento (Italia).

Una interesante aplicación de las técnicas de la fotogrametría y el uso del escáner láser es el que se muestra en el artículo denominado «3D Surveying & Modeling of Underground Passages in WWI Fortifications» (Rodríguez-Gonzálvez et al., 2015).

En el citado artículo se expone el procedimiento llevado a cabo para documentar estructuras subterráneas de la Primera Guerra Mundial ubicadas en Trento (Italia), con la dificultad que conlleva la falta de luz, y



Figura 1.20: A igualdad de resolución angular, la densidad de puntos varía en función a la inclinación de la superficie con respecto al instrumento (Nobile, 2012).



Figura 1.21: ZEB1 utilizado en el yacimiento arqueológico de Baia (Italia). Elaboración Propia.

la estrechez de las galerías.

Con este ejemplo, se intenta reflejar el interés de las técnicas de levantamientos, con el fin de preservar y documentar momentos históricos, no solo haciendo referencia a edificios, sino también a otro tipo de construcciones, como termas, galerías subterráneas o murallas, entre otros (Véase Figura 1.23).

Estos túneles fueron excavados para conectar fortificaciones adyacentes, y como pasajes de evacuación. En estos momentos, debido a colapsos, muchos de ellos se encuentran bloqueados, y necesitan ser restaurados para asegurar su debida preservación, e incluso permitir el acceso a visitantes. La obtención de un modelo tridimensional del estado actual de estas galerías se hace imprescindible.





Figura 1.22: ZEB-REVO con accesorios. Fuente: GeoSLAM.

El pasaje es de unos 100 metros de longitud, y una sección de 2x2 metros. El levantamiento se llevó a cabo con un Escáner Láser Leica HDS7000, y una estación total Topcon GPT7001. Se realizaron diecinueve estacionamientos, y se obtuvieron 285 millones de puntos, consiguiéndose una media de resolución de unos cuatro milímetros. El trabajo de Fotogrametría se realizó con una cámara Nikon D3x y se completó con 400 fotos (Véase Figura 1.24).

Los estacionamientos se realizaron cada ocho metros. Esa distancia fue elegida de acuerdo a la media de la sección del túnel, la cual afectaba a la distribución de la densidad de puntos, la precisión de los puntos (debido al gran ángulo de incidencia) y las oclusiones causadas por la tosca superficie.

- Reconstrucción de antigua muralla en Ontario, Canadá.

Otro ejemplo de aplicación de técnicas de levantamiento arquitectónico, es el narrado en el artículo «Restoring Fort Frontenac in 3D: Effective Usage of 3D Technology for Heritage Visualization» (Yabe et al., 2015). En él se describe cómo, mediante fotogrametría, se creó un modelo tridimensional de la parte que aún queda de una muralla situada en Ontario, Canadá, y que data de 1673 (Véase Figura 1.25). Gracias a la creación del modelo tridimensional, y con la colaboración de expertos arqueólogos, se ha logrado una reconstrucción de la fortaleza de forma virtual, con un escenario interactivo en el que el visitante puede moverse y acercarse a visualizar lo que le interesa.



Figura 1.23: Galería enterrada, Trento (Italia) (Rodríguez-González et al., 2015).

- Reconstrucción de las ruinas del Castillo de San Jorge, Sevilla.

Recientemente, como Trabajo Fin de Grado y en colaboración con la Universidad de Salerno (Italia), se ha realizado el trabajo de reconstrucción de las ruinas del Castillo de San Jorge, en Sevilla (Camillo Saviello et al., 2017). Mediante técnicas de escáner láser y de fotogrametría se realizó la toma de datos de las ruinas existentes, realizando un modelo tridimensional de las mismas.

Posteriormente, analizando la documentación gráfica de la que se disponía, se ha realizado una hipótesis de reconstrucción del castillo (Véase Figura 1.26). El modelo de la reconstrucción del castillo, junto con su entorno, se han integrado en el programa de realidad virtual *Unreal Engine*. Este *software* permite realizar una visita virtual por el interior del castillo (Véase Figura 1.27).

- Otros trabajos recientes.

Para continuar con recientes investigaciones en las que se aplican las técnicas de fotogrametría, cabe destacar el escenario virtual que se describe en el artículo «A Hand-Free Solution for the Interaction in An Immersive Virtual Environment: the Case of the Agora of Segesta» (Olivito et al., 2015). En este trabajo se ha conseguido una reconstrucción de una edificación del período Helenístico, mediante fotogrametría terrestre y aérea.

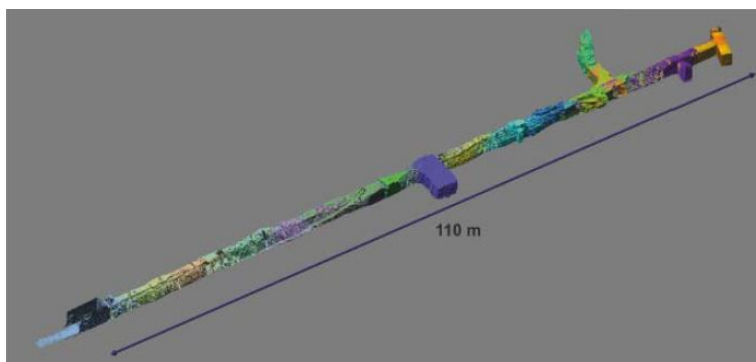


Figura 1.24: Nube de puntos de los 19 escaneos (Rodríguez-González et al., 2015).



Figura 1.25: Reconstrucción de la fortaleza (Yabe et al., 2015).

Por un lado, se ha creado un modelo tridimensional del estado actual, y por otro, una reconstrucción teórica de cómo se supone que debía ser. Ambos modelos se han integrado en uno solo, creando un escenario por el que el visitante puede caminar, y con gestos manuales puede ampliar la información y comparar ambos estados. Además, la iluminación se adapta al momento del día (Véase Figura 1.28).

### 1.5.2.4. Recomendaciones para la toma de datos

No son muchas las recomendaciones existentes con el fin de llevar a cabo una planificación de un trabajo de campo para la realización de un levantamiento arquitectónico.

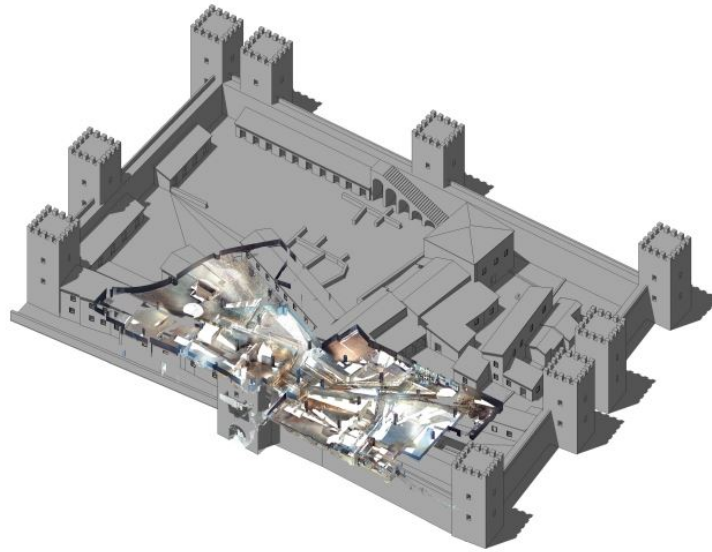


Figura 1.26: Hipótesis de reconstrucción del castillo mostrando las ruinas existentes integradas en él (Camillo Saviello et al., 2017).

Atendiendo a la obra *Il rilievo per il restauro* (Marino, 1990) las fases que han de preceder a los trabajos de documentación geométrica en un levantamiento son las siguientes:

- Reconocimiento del lugar de trabajo y valoración preliminar.
- Organización logística del equipo de trabajo.
- Señalización de puntos.
- Estimación y control de errores.
- Toma de datos, con las técnicas aplicables y diversa casuística.

Las recomendaciones más conocidas son las llamadas *Reglas 3x3* de la **CI-PA**, Comité Internacional de la Fotogrametría Arquitectónica. Presentadas por primera vez por Peter Whaldausl y Cliff Ogleby en 1994, en el V Simposio llamado *Close Range Techniques and Machine Vision* celebrado en Melbourne, Australia. Estas reglas conforman una guía para realizar una correcta toma de datos con cámara de fotos, es decir, usando la técnica de la Fotogrametría.



Figura 1.27: Vista del Castillo y su entorno simulando una reconstrucción de 1629 (Camillo Saviello et al., 2017).



Figura 1.28: Reconstrucción interactiva del *agora* (Olivito et al., 2015).

La última versión de estas reglas fueron publicadas en Noviembre de 2013, y han sido actualizada por Bil Blake, Andreas Georgopoulos y José Luis Lerma. En ellas se consideran los recientes desarrollos en adquisición de imágenes. Toda esta información puede encontrarse en el sitio web de la **CIPA** (CIPA, 2013).

Estas reglas son las siguientes:

### 1. Tres Reglas Geométricas.

#### a) De Control.

- 1) Medir algunas distancias entre puntos bien definidos.
- 2) Definir al menos una distancia vertical, usando una plomada o alguna característica vertical del edificio, y una distancia horizontal.
- 3) Hacer esto para todos los lados del edificio.
- 4) Si fuera posible, establecer una red de coordenadas tridimensionales con puntos de control que forme un circuito alrededor

del edificio.

b) Fotocubrición estereoscópica de amplio ángulo (Véase Figura 1.29).

- 1) Tomar las fotos formando un anillo alrededor del objeto, con un solape de al menos un 60 %.
- 2) Disparar desde una altura alrededor a la mitad del objeto, si es posible.
- 3) Incluir el contexto: línea de horizonte, cielo, etc.
- 4) Desde cada esquina del objeto, tomar una foto cubriendo los dos lados adyacentes.
- 5) Incluir la cubierta, si fuera posible.
- 6) No puede haber una imagen con falta de solape.
- 7) Añadir fotos ortogonales a las fachadas, para una vista previa y para su rectificación.

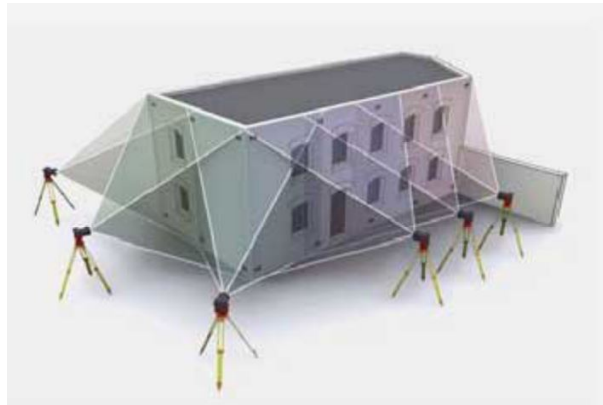


Figura 1.29: Fotocubrición Estereoscópica (CIPA, 2013).

c) Fotocubrición estereoscópica de detalle (Véase Figura 1.30).

Los pares estereoscópicos deben ser tomados de la siguiente forma:

- 1) Caso ortogonal: proporción base-distancia de 1:4 a 1:15.
- 2) Caso convergente: proporción base-distancia de 1:1 a 1:15.
- 3) Evitar el caso divergente.
- 4) Agregar cuadros cercanos para detalles y distancias de control, o añadir una cinta métrica en la vista. Comprobar los solapes de las fotografías.
- 5) En caso de duda, añadir más disparos y medir distancias para algún área potencialmente oscurecida.



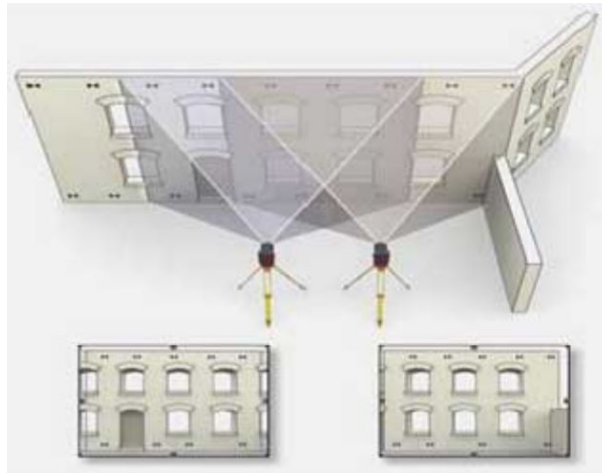


Figura 1.30: Par Estereoscópico (CIPA, 2013).

- 6) Asegurarse lo suficiente de que al menos cuatro puntos de control son visibles en el área de la estéreo imagen.

### 2. Tres Reglas de la Cámara.

#### a) Las propiedades de la cámara.

- 1) Fijar la óptica de la cámara. No hacer *zoom*. Deshabilitar el enfoque automático.
- 2) Fijar la distancia focal. Fijar al infinito, o una distancia determinada usando cinta adhesiva, pero solo usar una distancia para las fotos del anillo, y otra distancia para las fotos en detalle.
- 3) El marco de formato de imagen de la cámara debe ser claramente visible en las imágenes y tener un buen contraste.
- 4) Los documentos verdaderos son las diapositivas originales, negativos o sus equivalentes digitales, es decir, la imagen en formato *RAW*. Configurar la cámara para utilizarla en su formato de máxima calidad.

#### b) Calibración de la cámara.

Usar la mejor calidad, la resolución más alta y el mayor formato de cámara disponible.

- 1) El formato mediano es mejor que el formato pequeño. Un sensor grande es mejor que un sensor pequeño.
- 2) Una lente de ángulo amplio es mejor que una lente de ángulo estrecho. Evitar los objetivos gran angular.

- 3) Calibrar la cámara con el enfoque fijo.
  - 4) La información estándar de calibración es necesaria para cada combinación lente/cámara y para cada ajuste del enfoque que se use.
  - 5) Se debe utilizar una tabla de colores estandarizada en cada secuencia de cuadros.
- c) Exposición de la imagen.
- Se requiere una exposición consistente.
- 1) Trabajar con iluminación consistente: cuidado con las sombras oscuras y profundas.
  - 2) Usar HDR para capturar exposiciones difíciles y no equilibradas.
  - 3) Planear el mejor momento del día.
  - 4) Usar un trípode y un cable de control remoto para evitar el movimiento de la cámara y conseguir imágenes definidas.
  - 5) Utilizar una cabeza panorámica del trípode para obtener imágenes panorámicas sin paralaje.
  - 6) Utilizar el medio adecuado: el blanco y el negro son suficientes para trazar líneas, pero el color tiene algunas ventajas para la interpretación y la documentación de colores.
  - 7) Utilizar el formato RAW o alta calidad y alta sensibilidad en cámaras digitales.
  - 8) Se recomienda geoetiquetar las imágenes.

### 3. Tres Reglas de Procedimiento.

- a) Diseño del registro de las fotos.
- Elaborar documentación gráfica donde se indique lo siguiente:
- 1) Un plano del sitio, indicando la dirección del norte.
  - 2) Los alzados de cada fachada, a una escala apropiada, 1:50, 1:100, 1:500.
  - 3) Localización de las fotos y la dirección de las mismas, con la numeración de cada imagen.
  - 4) Cubrición por fotogrametría monoscópica y estereoscópica.
  - 5) Localización de los puntos de control, distancias y líneas de plomadas.
  - 6) Si se usan dianas naturales, mostrar en un diagrama claro los puntos tomados.



b) Registrar los datos.

Incluir lo siguiente:

- 1) El nombre del sitio, la localización y la georreferenciación, el nombre del propietario y la dirección.
- 2) Fecha, tiempo y personal.
- 3) Cámaras usadas, lentes y ajustes.
- 4) Informe de calibración.
- 5) Descripción del lugar, del sitio, de la historia y la bibliografía.
- 6) Documentación del proceso de trabajo.

c) Archivo.

Los datos adquiridos deben estar completos, estables, seguros y accesibles.

- 1) Comprobar si algo no está completo o correcto antes de abandonar el lugar.
- 2) Guardar las imágenes en una ubicación segura de la cámara. Guardar las imágenes en formato *RAW* y convertirlos en archivos *TIFFs*.
- 3) Escribir todo el trabajo realizado inmediatamente.
- 4) Los negativos originales son documentos para archivar y se deben tratar y conservar cuidadosamente.
- 5) No cortar el formato original.

Estas Reglas 3x3 son aplicables solamente a trabajos de fotogrametría. A parte de estas recomendaciones, se pueden consultar algunas más en el libro *Fundamentos Teóricos de los Métodos Topográficos* (Ríos, 2000), como las siguientes:

- Proyectar la triangulación con triángulos lo más equiláteros posibles.
- Evitar desigualdades excesivas entre las longitudes de los lados.
- Evitar ángulos en vértices inferiores a 25 grados centesimales.

Por otro lado, en el manual *Construction Surveying and Layout* (Crawford, 1995), pueden encontrarse algunas recomendaciones referentes a la organización del trabajo. Éstas se limitan a la planificación de un horario de trabajo, y del uso del croquis. También hace hincapié en la comprobación *in situ* de la visualización que proporciona un determinado emplazamiento.

Para continuar con la revisión de las distintas recomendaciones, pueden consultarse las indicadas en el artículo «Aplicaciones de geotecnologías láser y fotogramétricas a la digitalización de escenarios complejos» (de Frutos et al., 2011), para la planificación de los estacionamientos desde los cuales realizar un levantamiento arquitectónico con escáner.

- Las fachadas deben quedar escaneadas desde al menos una estación.
- Las tomas deben evitar la oblicuidad con la geometría del objeto, es decir, deben ser frontales.
- Se necesita un recubrimiento mínimo entre escaneos de distintas bases del 10 %.
- Se deben evitar el mayor número de sombras en la nube de puntos.
- El número total de escaneos no ha de ser excesivo por motivo de aumento del trabajo del procesado y la acumulación por arrastre de errores.

Por último, en el libro *Metric Survey Specifications for Cultural Heritage* (Bryan et al., 2013) se especifica de forma detallada cada una de las fases de las que se compone el trabajo de levantamiento arquitectónico, diferenciando entre el levantamiento realizado con escáner, con estación total y con cámara de fotos. Para cada uno de ellos se dan una serie de recomendaciones, relativas a la prevención y los riesgos de los instrumentos, de la planificación del trabajo, del nivel de detalle, la escala, y muchas otras recomendaciones. Sin embargo, en ninguna de ellas se establece un método para llevar a cabo de forma correcta la adquisición de datos.

#### **1.5.2.5. Del Diseño de Redes en Fotogrametría Terrestre**

En diferentes artículos se hace referencia a la importancia del diseño de una red óptima de fotogrametría (Parian and Gruen, 2005) (Kiamehr, 2003). Diseñar una red de fotogrametría terrestre consiste en establecer la configuración de un itinerario y un plan de observación óptimos que satisfacerá la calidad del conjunto con el mínimo esfuerzo (Parian and Gruen, 2005).

En la actualidad, la obtención de un modelo tridimensional se realiza, principalmente, usando técnicas de Modelado Basado en Imágenes (*IBM techniques*) (Alsadik et al., 2012).

Durante las tres últimas décadas se ha tratado de resolver el problema del **Diseño de una Red Fotogrametría** (*Photogrammetric Network Design* - PND) usando diferentes técnicas matemáticas. Destacan principalmente los siguientes investigadores: Andrew R. Marshall, Erik Grafarend, Clive Fraser, Scott Mason, Gustavo Olague y Mohamad Saadatseresht, entre otros.

**Principios de Diseño de una Red de Fotogrametría** Las características que ha de tener una Red de Fotogrametría para que sea óptima son las siguientes: Precisión, Exactitud, Fiabilidad, Sensibilidad y Economía (Marshall, 1989).

1. Precisión → Se trata del grado de conformidad entre un conjunto de observaciones de la misma variable aleatoria. Se estima analizando la Matriz de Varianza-Covarianza<sup>4</sup>.
2. Exactitud → Mide la proximidad de un valor a su valor verdadero.
3. Fiabilidad → La fiabilidad es la forma de medir el potencial que presenta una red para detectar errores groseros y sistemáticos.
4. Sensibilidad → Es la capacidad de una red para detectar movimiento en un punto. Es aplicable en análisis de deformación, donde las observaciones se miden en dos momentos diferentes.
5. Economía → Gastos en establecer, medir y evaluar una red.

**ZOD FOD SOD TOD** Durante años se ha considerado imposible resolver el problema de diseñar una red fotogramétrica en un solo paso, por lo que este problema fue dividido en cuatro pasos propuestos por Erik Grafarend (Grafarend, 1974). Estos cuatro problemas son los siguientes:

- **ZOD** (*Zero-Order Design*) → “The Datum Problem” (Problema de Orden Cero: Problema del Datum). Consiste en la elección de un sistema de referencia óptimo para las coordenadas espaciales del objeto. Para ello se hace uso de la Matriz de Varianza-Covarianza.

---

<sup>4</sup>A través de la Matriz Varianza-Covarianza se puede comprobar la precisión de las coordenadas del punto del objeto. Geométricamente, cada Matriz de Covarianza representa un elipsoide, el cual varía en orientación y tamaño para cada posición de la cámara. El error elipsoidal indica la precisión con la que se están suponiendo las coordenadas de un punto de un objeto en su modelo tridimensional

- **FOD** (*First-Order Design*) → “The Configuration Problem” (Problema de Primer Orden: Problema de la Configuración). Resolviendo este problema se obtendrá una configuración adecuada de la cámara: distancia focal, sistema de medida de la imagen, y una primera aproximación de la geometría adecuada de la red (Olague, 2001). Se consigue el posicionamiento óptimo de los puntos y el diseño de un plan de observación óptimo.
- **SOD** (*Second-Order Design*) → “The Weight Problem” (Problema de Segundo Orden: Problema del Peso). En este problema se estudia la precisión óptima y la distribución de las observaciones.
- **TOD** (*Third-Order Design*) → “The Densification Problem” (Problema de Tercer Orden: Problema de Densificación). Con este último paso, se mejora la red mediante la adición de puntos de observación.

A pesar de que estos problemas fueron propuestos en 1974, a día de hoy se siguen teniendo en cuenta. Estos sub-problemas sirven para estudiar la calidad del diseño de la red. Hay que tener en cuenta que estas pautas están interconectadas, por lo que para alcanzar la solución no basta con solucionar paso a paso estos problemas (Fraser, 1984).

De estos cuatro pasos, los tres primeros son los que se aplican a la Fotogrametría de rango cercano, (Marshall, 1989).

Las restricciones más importantes que definen una buena solución al PND (*Photogrammetric Network Design*) fueron propuestas por Mason en 1994 (Mason, 1994).

Cada cámara deberá ser estacionada de tal forma que satisfaga unas restricciones básicas:

- Escala de la imagen.
- Resolución.
- Espacio de trabajo.
- Profundidad de campo.
- Ángulo de incidencia.
- Número y distribución de puntos de la imagen.

- Iluminación.

Además de estas siete restricciones, existen unos objetivos que deben ser considerados a la hora de ubicar las cámaras.

- Contribución a la intersección de los ángulos.
- Campo de visión.
- Visibilidad.

### 1.5.3. De los Métodos de Optimización en la Ingeniería

Tomando las palabras de Singiresu S. Rao (Rao, 2009), la **Optimización** se define como el acto de obtener el mejor resultado en determinadas circunstancias. En el campo de la Ingeniería, se deben tomar decisiones que supongan el mínimo esfuerzo requerido para maximizar el beneficio deseado.

Puesto que tanto el esfuerzo requerido como el beneficio deseado en cualquier situación práctica se puede expresar como una función de ciertas variables de decisión, la optimización se puede definir como el proceso de encontrar las condiciones que dan el valor máximo o mínimo a dicha función. No hay un único método para resolver todos los problemas de optimización.

Cuando se abordan problemas cuyo espacio de posibles soluciones es reducido, la búsqueda consiste en hallar un valor óptimo (un máximo o un mínimo) que dé solución al problema, dadas unas restricciones. Sin embargo, existen problemas de optimización combinatoria complejos, cuyo espacio de búsqueda es muy extenso, por lo que resultan imposibles de resolver de una forma sencilla. Esto ocurre en diversos campos, como la Economía, el Comercio o la Ingeniería (Suarez, 2013).

Como se ha comentado en la Sección 1.4, los problemas altamente complejos, es decir, difíciles de resolver, son denominados **Problemas NP**. Son problemas para los cuales no es posible garantizar que se pueda encontrar una mejor solución en un tiempo polinómico reducido.

Se diferencia entre dos tipos de algoritmos, en función a la dimensión del campo de búsqueda de soluciones:

- **Algoritmos exactos.** Solo es posible llevarlo a cabo cuando el campo de búsqueda es reducido, ya que se recorren todas las posibles soluciones.

- **Algoritmos aproximados, Heurísticas y Metaheurísticas.** Cuando el espacio de búsqueda es amplio, se trata de buscar una solución aproximada y razonable en un tiempo reducido.

#### 1.5.3.1. Heurísticas

El término **Heurística** proviene de la Grecia clásica, originado de la palabra griega *heuriskein*, y cuyo significado es *encontrar* o *descubrir*. Deriva de la famosa exclamación atribuida a Arquímedes: *eureka!*.

Al hablar de Algoritmos Heurísticos o simplemente Heurísticas, se hace referencia a un algoritmo que puede producir una buena solución a un problema, aunque no se puede asegurar que sea la óptima ya que no se han recorrido todas las soluciones posibles.

Las técnicas heurísticas son procesos eficientes conscientes o inconscientes los cuales ignoran parte de la información, para, de esta forma, ser capaces de tomar decisiones de una forma más rápida. Usar heurísticas supone ahorrar esfuerzo. Comúnmente se ha entendido que las decisiones heurísticas implican mayores errores que tomar decisiones racionales, basadas en la lógica o la estadística. Sin embargo, para muchas situaciones, el asumir decisiones de forma racional no es posible (Gigerenzer and Gaissmaier, 2011).

Cuando las heurísticas fueron formalizadas se descubrió que en muchos casos las heurísticas simples eran más precisas que los métodos estadísticos estándar. Llega un momento en el que más información no es mejor, sino que es perjudicial para tomar una decisión, a este efecto se le conoce como *menos-es-más*.

Entre los Algoritmos Heurísticos más utilizados, destacan los siguientes:

- **Algoritmos iterativos.** A partir de una solución factible se construye otra mejor, y mediante iteraciones de este proceso se mejoran las soluciones, hasta verificar cierto criterio de parada.
- **Algoritmos voraces - Greedy.** Los métodos voraces van construyendo paso a paso la solución, de manera que hasta el final no se conoce la solución de forma factible.

El nombre que recibe este algoritmo viene justificado de la siguiente forma: En cada iteración, el algoritmo selecciona el mejor bocado que puede tragar, sin preocuparse por el futuro. Nunca cambia de opinión: una vez

que un candidato se ha incluido en la solución, queda allí para siempre; una vez que se excluye un candidato de la solución, nunca vuelve a ser candidato.

- **Algoritmos Probabilistas.** Entre ellos están los Algoritmos Numéricos, Algoritmos de Monte Carlo y el Algoritmo de las Vegas.

Cuando se resuelve un problema por métodos heurísticos, ya que no es posible garantizar la optimalidad del mismo, es necesario medir la calidad de los resultados. Para tener un buen algoritmo heurístico debe ser eficiente, robusto y bueno.

### 1.5.3.2. Inteligencia Artificial y Metaheurísticas

#### Inteligencia Artificial.

Se hace a continuación una semblanza general de la **Inteligencia Artificial**, en lo que sigue IA, según el profesor Sancho Caparrini (Sancho Caparrini, 2015), con el fin de enmarcar dentro de ella los algoritmos usados en los métodos que se proponen en esta tesis.

La IA es el estudio de la conducta inteligente. Los objetivos que persigue el estudio de la IA son dos: el estudio del proceso de pensamiento y la conducta inteligente de los humanos; y producir máquinas o sistemas que representen estos procesos (Mendez et al., 2007).

El campo de la Inteligencia Artificial clásica aparece en los años 50, como resultado de la comprensión del funcionamiento del cerebro, a través de la neurociencia y gracias a la aparición del ordenador, entre otras razones. La IA está considerada como una disciplina de las Ciencias de la Computación, y se trata de un campo transdisciplinar.

Dentro de la IA se encuentra el área de la **Inteligencia Computacional**, la cual describe técnicas que se centran en la obtención de estrategias y resultados. Este campo incluye las siguientes sub-áreas:

- Computación Evolutiva.
- Inteligencia de Enjambre.
- Redes Artificiales Neuronales.

- Inteligencia Difusa.
- Sistemas Inmunes Artificiales.

### **Metaheurísticas.**

La **Metaheurística** es otra manera de denominar formas de obtener estrategias en IA. Como se ha mencionado anteriormente, una heurística es un algoritmo que localiza soluciones suficientemente buenas para un problema. Los métodos heurísticos compensan la precisión y calidad de las soluciones con la eficiencia computacional, en espacio y tiempo.

Los Procedimientos Metaheurísticos, o Metaheurísticas son una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que las heurísticas clásicas no son ni efectivas ni eficientes. Las metaheurísticas proporcionan un marco general para crear nuevos algoritmos híbridos, combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos (Osman and Kelly, 1996). La diferencia entre las heurísticas y las metaheurísticas es que las segundas combinan uno o más métodos heurísticos, usando una estrategia de nivel superior.

Las propiedades de las técnicas metaheurísticas son las siguientes (Sancho Caparrini, 2009):

1. Son estrategias que guían el proceso de búsqueda.
2. El objetivo es explorar eficientemente el espacio de búsqueda con el fin de encontrar soluciones (sub)-óptimas.
3. Las técnicas que constituyen los algoritmos de metaheurística van desde simples procedimientos de búsqueda local hasta complejos procedimientos de aprendizaje.
4. Los algoritmos metaheurísticos suelen ser aproximados y, habitualmente, no deterministas.
5. Pueden incorporar mecanismos para evitar quedarse confinados en áreas limitadas del espacio de búsqueda.
6. No son específicos para un problema.
7. Pueden hacer uso de conocimiento específico del dominio en la forma de heurísticas que son controladas por estrategias de nivel superior.



8. Las metaheurísticas más avanzadas hacen uso de experiencias de búsqueda (en forma de memoria) para guiar las búsquedas.

En un siguiente nivel se encontrarían las **Técnicas Hiperheurísticas**, las cuales proporcionan estrategias al nivel más alto, pudiendo emplear aprendizaje automático y adaptar su comportamiento de búsqueda.

A continuación se describen las principales técnicas metaheurísticas:

- **Recocido Simulado** - *Simulated Annealing*. Se basa en determinados principios de la termodinámica. El término *recocido* es utilizado en mecánica de materiales para referirse al proceso de enfriar un material lentamente hasta que llegue a un estado estable. Al comienzo de la búsqueda, la temperatura es elevada, por lo que el algoritmo ofrece cierta flexibilidad para moverse a una peor solución. Pero, a medida que el material se enfría, dicha flexibilidad se va reduciendo, de forma que el valor de la función objetivo se establece al final del proceso (Páez Parra and Vargas Méndez, 2013).
- **Optimización por Enjambre de Partículas. PSO** - *Particle Swarm Optimization*. Motivada por el comportamiento social de una manada de pájaros. Cada partícula ajusta su *vuelo* de acuerdo a su posición y experiencia de vuelo.
- **Algoritmos Genéticos**. Son algoritmos de búsqueda y optimización computarizados, basados en la mecánica de la genética natural y la selección natural. Fueron propuestos originalmente por John Holland en 1975. Entre las técnicas evolutivas, los Algoritmos Genéticos son el grupo más extendido (Sivanandam and Deepa, 2007, p.V).
- **Búsqueda Tabú**. Una vez que una potencial solución es determinada, se la marca como *tabú* de modo que el algoritmo no vuelva a visitar esa posible solución, explorando así soluciones más allá del óptimo local.
- **Colonia de Hormigas** - *Ant System*. Simula el comportamiento de colonias de hormigas, las cuales, guiadas por las feromonas, son capaces de adaptarse al medio para conseguir encontrar el camino más corto. Este comportamiento es llevado a colonias de hormigas artificiales, para la búsqueda de soluciones aproximadas. El método fue desarrollado por primera vez por Marco Dorigo en 1992 (Véase Figura 1.31).

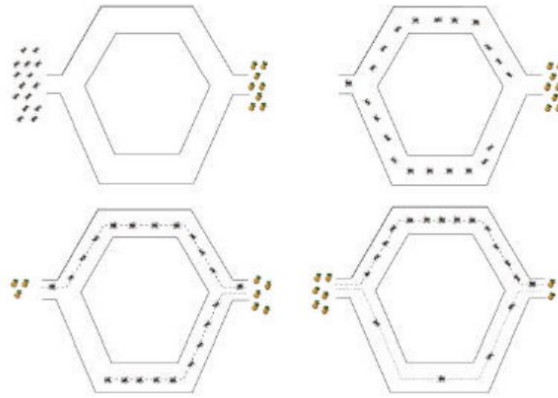


Figura 1.31: El comportamiento de la colonia termina por obtener el camino más corto entre dos puntos (Sancho Caparrini, 2015).

- **Procedimiento GRASP** - *Greedy Randomized Adaptive Search Procedure*. Método iterativo que construye soluciones de alta calidad que son procesadas posteriormente para obtener otras soluciones aún mejores.

En la Figura 1.32 puede verse la distribución de publicaciones recientes utilizando las diferentes técnicas metaheurísticas: Algoritmos Genéticos, GA (*Genetic Algorithms*); Optimización por Enjambre de Partículas, PSO (*Particle Swarm Optimization*); Optimización por Colonia de Hormigas, ACO (*Ant Colony Optimization*); Algoritmo de Búsqueda Gravitacional, SA (*Gravitational Search Algorithm*), Búsqueda Tabú, TS (*Tabu Search*); Programación Evolutiva, EP (*Evolutionary Programming*); y Búsqueda de Armonía, HS (*Harmony Search*).

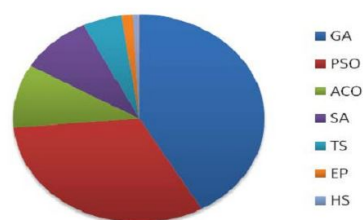


Figura 1.32: Distribución de publicaciones científicas usando las distintas metaheurísticas (Fawzy, 2015, p.148).

### 1.5.3.3. Aplicaciones de la Optimización en Problemas de Ingeniería

Algunas de las aplicaciones de los Métodos de Optimización en el campo de la Ingeniería son los siguientes (Rao, 2009):

- Diseño de estructuras aeronáuticas y aeroespaciales para el peso mínimo.
- Encontrar las trayectorias óptimas de vehículos espaciales.
- Diseño de estructuras de ingeniería civil, como puentes, torres, chimeneas o presas a coste mínimo.
- El peso mínimo de las estructuras de sismo, viento, y otros tipos de carga aleatoria.
- Diseño de sistemas de recursos hídricos para el máximo beneficio.
- El diseño óptimo de estructuras de plástico.
- Selección de condiciones de mecanizado en procesos de corte de metal para la producción al mínimo coste.
- Diseño de equipos de manejo de materiales, tales como transportadores, camiones y grúas, por un coste mínimo.
- Diseño de bombas, turbinas y equipos de transferencia de calor para la máxima eficiencia.
- Diseño óptimo de maquinaria eléctrica, tales como motores, generadores y transformadores.
- Diseño óptimo de redes eléctricas.
- Ruta más corta tomada por un vendedor para visitar varias ciudades durante una gira.
- Óptima planificación de la producción, control y programación.
- Diseño óptimo de equipos de procesamiento químico y plantas.
- Diseño óptimo de redes de tuberías para industrias de procesos.
- Emplazamiento para una industria.
- Control de inventario.
- El control de los tiempos de espera y de inactividad y las colas en las líneas de producción para reducir los costes.

#### 1.5.3.4. Algoritmos Genéticos aplicados al Diseño de Redes de Fotogrametría

De entre las distintas técnicas de optimización metaheurísticas, este trabajo de investigación se centra en los **Algoritmos Genéticos**.

Haciendo uso de la definición para el término dada por Abdelmalik Moujahid, Iñaki Inza y Pedro Larrañaga: «Los Algoritmos Genéticos son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acuerdo con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin en 1859. Por imitación de este proceso, los Algoritmos Genéticos son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de las mismas.» (Moujahid et al., 2008).

Los Algoritmos Genéticos han sido usados para solucionar problemas complejos, como problemas *NP-Duros*; e incluso para desarrollar programas simples. Algunas de las aplicaciones son las siguientes (Sivanandam and Deepa, 2007).

- Planificación de la trayectoria de robots.
- Planificación de estrategias.
- Búsqueda de la forma de proteínas moleculares.
- Funciones de creación de imágenes.
- Control de tuberías de gas.

Los Algoritmos Genéticos están siendo aplicados a todo tipo de problemas de optimización, el Problema del Diseño de Redes de Fotogrametría (PND), sobre el cual se ha escrito en el Apartado 1.5.2.5, entre ellos.

Gustavo Olague en sus artículos «Optimal Camera Placement to Obtain Accurate 3D Point Positions» (Olague and Mohr, 1998), «Design and simulation of the Photogrammetric Networks using Genetic Algorithms» (Olague et al., 2000) y «Automated Photogrammetric Network Design Using Genetic Algorithms» (Olague, 2001), diseña un algoritmo genético mediante el cual

## 1.5. ESTADO DE LA CUESTIÓN

minimiza la varianza media de los puntos tridimensionales de un objeto, satisfaciendo unas restricciones dadas.

Este algoritmo genético minimiza el error elipsoidal de los puntos de un objeto. El error elipsoidal indica la precisión con la que se están suponiendo las coordenadas de un punto de un objeto en su modelo tridimensional (Véase Figura 1.33).

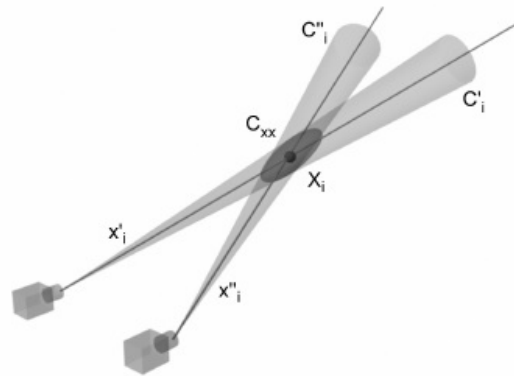


Figura 1.33: La medida del eje menor y del eje mayor del elipsoide que se forma, es una medida de calidad de la geometría de la escena (Beder and Steffen, 2006).

A través de la Matriz de Varianza-Covarianza se puede comprobar la precisión de las coordenadas del punto del objeto. Geométricamente, cada matriz representa un elipsoide, el cual varía en orientación y tamaño para cada posición de la cámara.

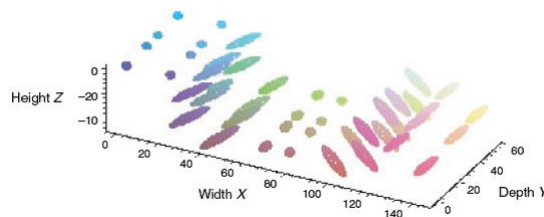


Figura 1.34: Representación de Error Elipsoidal (Olague, 2001).

Los algoritmos genéticos son técnicas iterativas de optimización. Las iteraciones finalizarán cuando se dé alguna de las siguientes condiciones de parada:

- Cuando se llegue a un número máximo establecido de iteraciones.
- Cuando los resultados no mejoren después de un número determinado de iteraciones.

Partiendo de un conjunto de posibles soluciones, mediante la comparación entre las mismas, se evalúa cuál es la más óptima. Las demás soluciones son soluciones alternativas.

La evaluación de una red fotogramétrica está directamente relacionada con el ángulo de separación entre cada cámara que compone la red. Una vez que se ha encontrado una solución óptima, es decir, una buena distribución de las cámaras, si se realizan pequeños desplazamientos, la precisión del sistema no se verá afectada. Además el algoritmo debe tener en cuenta una serie de restricciones, las cuales se enumeran a continuación.

1. Contribución a los ángulos de intersección entre cámaras.
2. Ángulo de convergencia entre la cámara y el objeto, estableciéndose un ángulo mínimo de incidencia admisible.
3. Restricciones del espacio de trabajo, debido a obstáculos, muros o cualquier otra obstrucción.
4. Visibilidad.

Para facilitar la comprensión del funcionamiento de este Algoritmo Genético, a continuación se detalla cómo funciona el ejemplo descrito en la Figura 1.35:

1. Se parte de una población aleatoria de  $N = 30$  redes fotogramétricas convergentes, que satisfagan la restricción medioambiental. Cada cámara  $Cam_m$  es codificada por dos parámetros  $(\alpha, \beta)$ , en código binario y en cadena.
2. A continuación, se evalúa cada red fotogramétrica completa, y se almacena la varianza media en la matriz  $\sigma_c^2$ . Este valor indica cómo de buena es una solución con respecto a otras soluciones de la población  $P(t)$  en la generación  $t$ .
3. En este momento, se selecciona un conjunto de redes buenas, mediante selección por torneo: se seleccionan dos redes de  $P(t)$  y la mejor, de acuerdo con la Función Objetivo, pasa a la siguiente generación  $P(t+1)$ .

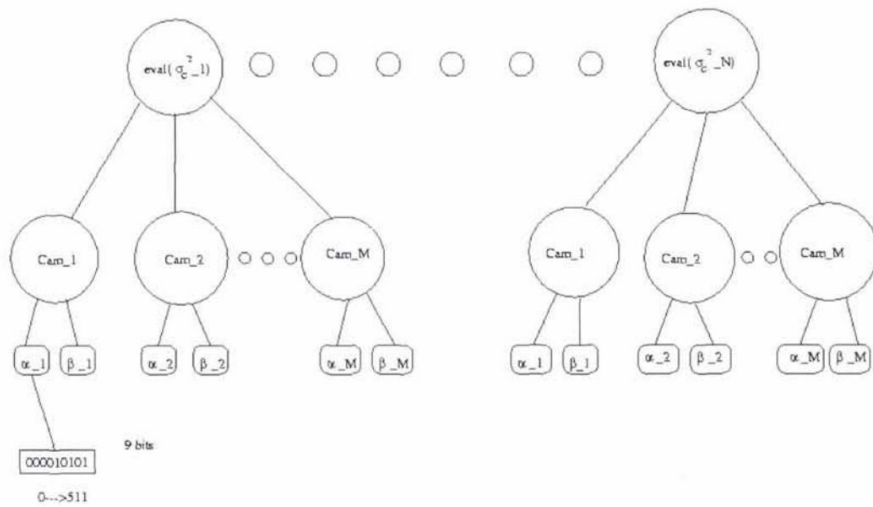


Figura 1.35: Esquema de Algoritmo Genético (Olague, 2001, p.428).

4. Con esta nueva población, se recombinan las cadenas binarias para cada cámara utilizando las siguientes operaciones:
  - a) Cruce, con una probabilidad  $P_c = 0,7$ .
  - b) Mutación, con una probabilidad  $P_m = 0,005$ .
5. Los pasos 2, 3 y 4 se repiten hasta que los criterios de optimización se estabilicen.

Posteriormente, el mismo autor desarrolla un método para cambiar la posición de los puntos de vista, de forma que se reduzca el error elipsoidal (Olague and Dunn, 2007). Para una mayor comprensión véase la Figura 1.36.

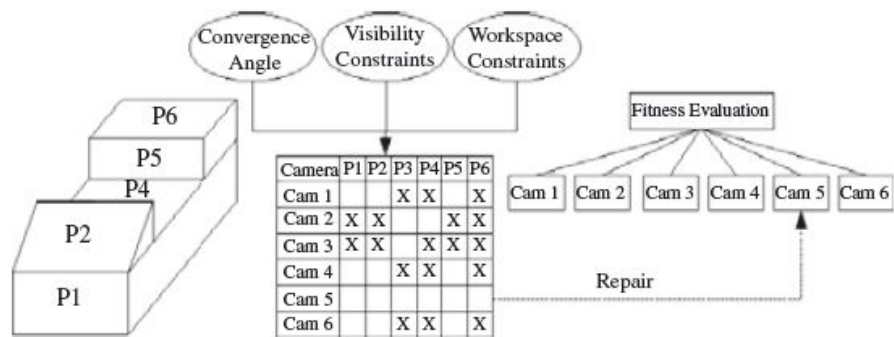


Figura 1.36: Evaluación de una red de fotogrametría (Olague and Dunn, 2007).



### 1.5.4. Investigaciones actuales relacionadas con el problema planteado

En este apartado se describen diferentes algoritmos desarrollados recientemente relacionados con el problema que se propone: optimizar los puntos de vista desde los cuales cubrir la vigilancia de un objeto. Todos ellos han servido como inspiración para diseñar los algoritmos que se proponen en esta tesis. Estos algoritmos se exponen de acuerdo al año de publicación de los mismos.

#### 1.5.4.1. Algoritmo I

Resulta interesante para este trabajo de investigación la apreciación que puede leerse en el artículo «Viewpoint Selection for Complete Surface Coverage of Three Dimensional Objects» (Roberts and Marshall, 1998). De ahí puede traducirse que determinar los puntos de vista mejora la realización de tareas como la inspección de un objeto, reduciendo la cantidad y la redundancia en la adquisición de datos, así como el tiempo de captura de los mismos.

En este artículo se describe un algoritmo diseñado para la vigilancia de objetos tridimensionales mediante grafos. En estos grafos, cada cara del objeto es representada por un nodo. Las aristas del grafo unen aquellas caras que en el objeto real comparten un lado. De esta forma es fácil agrupar las caras de un objeto (Véase Figura 1.37).

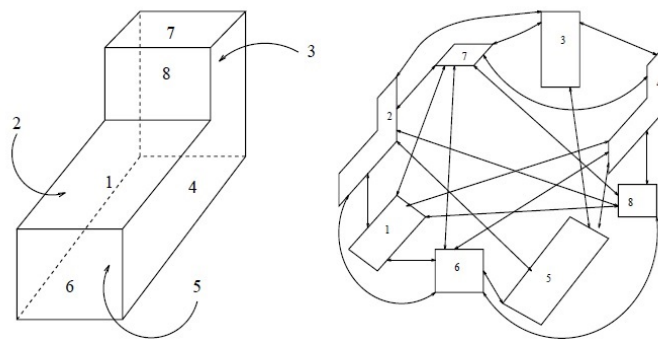


Figura 1.37: Objeto tridimensional y su correspondiente grafo (Roberts and Marshall, 1998).

El algoritmo consiste en buscar un punto de vista que vigile al mayor grupo de caras que pueden ser vistas simultáneamente. A continuación se buscará el

punto de vista que vigile al segundo mayor grupo de caras que puedan ser vistas de forma simultánea. Y de esta forma se irán consiguiendo los puntos de vista para vigilar todo el objeto (Véase Figura 1.38).

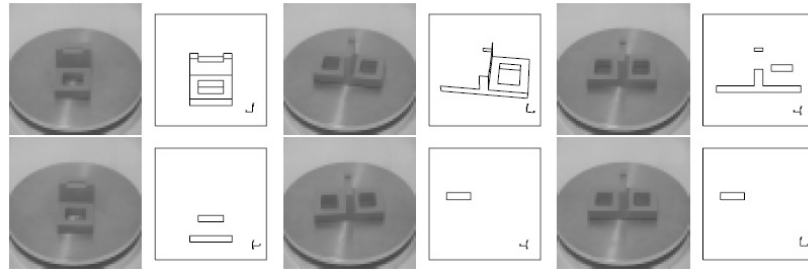


Figura 1.38: Vistas reales y esquema de las caras vigiladas para cada grupo. Sólo las caras que faltan por ser vistas son representadas en el esquema (Roberts and Marshall, 1998).

Unos años después, y siguiendo en esta misma línea, en el artículo «Viewpoint Selection using Viewpoint Entropy» (Vázquez et al., 2001) se habla sobre el concepto de **Entropía del Punto de Vista**. Esta nueva unidad de medida trabaja sobre las áreas proyectadas de las caras de un objeto, y mide la cantidad de información capturada desde un punto. La calidad del punto de vista está relacionada con la cantidad de información que éste proporciona sobre una escena.

Como puede leerse en el artículo anteriormente citado, la computación de un buen punto de vista es importante en diferentes campos de investigación, como la geometría computacional, el control servo-visual, los movimientos de robot, o el dibujo gráfico, entre otros.

El valor de esta medida permite computar buenas posiciones de forma automática. El mejor punto de vista será aquel que mayor entropía tenga (Véase Figura 1.39).

La entropía de un punto de vista tiene en cuenta dos variables: el área proyectada, y el número de caras que son vistas.

En la investigación a la que se hace referencia, se ha diseñado un algoritmo y se ha implementado usando *OpenGL*. El objetivo del algoritmo es seleccionar el conjunto de puntos de vista mínimos que proporcione una buena representación de la escena al usuario de un objeto con un número determinado de caras (Véase Figura 1.40).

El algoritmo convergerá cuando un porcentaje elevado de caras del objeto

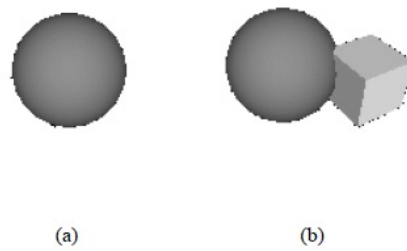


Figura 1.39: (a) Un mal punto de vista y (b) un buen punto de vista de una escena que contiene una esfera y un cubo (Vázquez et al., 2001).

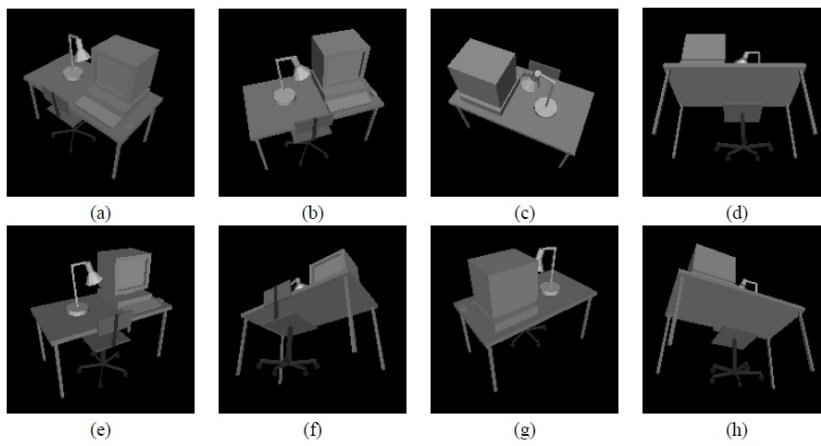


Figura 1.40: Los ocho puntos de vista obtenidos para visualizar la escena (Vázquez et al., 2001).

haya sido visto.

De este estudio se sacan las siguientes conclusiones para aplicar al algoritmo que se diseña en la presente tesis:

- La entropía de un punto, es decir, la cantidad de información que se obtiene desde una posición, es un factor imprescindible a tener en cuenta.
- Es necesario buscar aquellos puntos con mayor entropía para conseguir una vigilancia óptima de un objeto.
- La idea es la de encontrar el punto con mayor entropía para vigilar un objeto.

- Una vez obtenido dicho punto, se trata de encontrar el siguiente punto con mayor entropía que vigile las partes del objeto que no hayan sido vigiladas anteriormente, y así sucesivamente.

#### 1.5.4.2. Algoritmo II

En los últimos años, una nueva técnica de adquisición de imágenes ha cobrado gran importancia, la cual funciona de forma automatizada y para la que la altura y la posición no es una limitación a la hora de realizar una toma de datos fotográficos. Se trata de los **Vehículos Aéreos No Tripulados**, también conocidos como *drones*. Un grupo de investigación de Austria, ha centrado sus estudios en un tipo de drones, en concreto los de reducido tamaño: los **Micro Vehículos Aéreos No Tripulados**, *Micro Aerial Vehicles* (MAVs) (Véase Figura 1.41).



Figura 1.41: Micro dron md4-200 con cámara anexa (Irschara et al., 2010, p.65).

Trabajar con este tipo de instrumentos abre nuevas posibilidades, al ofrecer mayor flexibilidad. Es posible crear redes múltiples y simultáneas, así como trabajar con objetos que no tienen porqué ser rígidos (Irschara et al., 2010).

A la hora de diseñar una red, además de conocer la ubicación desde las cuales se tomarán las fotografías, es importante realizar un estudio del recorrido que se llevará a cabo. Sobre todo, si el trabajo se va a realizar con un micro vehículo aéreo, dado que el tiempo de su batería es bastante limitado: de entre diez a veinte minutos.

El algoritmo propuesto por este grupo de investigación en su artículo denominado «Photogrammetric Camera Network Design for Micro Aerial Vehicles» (Hoppe et al., 2012), para diseñar una red de fotogrametría, tiene en cuenta las siguientes restricciones:

1. El número de posicionamientos debe ser pequeño.

2. Deben existir redundancia y solape suficientes en las imágenes tomadas.
3. Se debe tener en cuenta el ángulo de visión entre fotos que presentan zonas solapadas.
4. El grafo de solape de imagen debe estar totalmente conectado, de otra forma, la reconstrucción no quedaría unida.

Como se puede leer en el artículo anteriormente mencionado, para calcular los puntos de vistas para diseñar una red que cubra a un edificio, es necesario conocer previamente la geometría del objeto. Es decir, sería necesario partir de un modelo tridimensional aproximado. Por eso se describe a este problema como *el Problema de la gallina y el huevo*: si para planificar el proceso de toma de datos de un edificio se debe partir de la forma del edificio, ¿cómo obtener el modelo previo? La respuesta es que se trata de mejorar un modelo del que se pueden tener algunos datos, un modelo en bruto.

El **Problema del Conjunto de Cobertura**, *Set-Cover Problem*, es un problema *NP*-completo. Añadiendo algunas restricciones, como el solape o el ángulo máximo de las triangulaciones, el problema se convierte en un **Problema de Satisfacción de Restricciones** (CSP). Este tipo de problemas es normalmente resuelto con algoritmos de búsqueda.

Teniendo en cuenta estas restricciones, se diseña un algoritmo que consta de los siguientes pasos:

1. Emplazamiento de la cámara.

A partir del modelo aproximado  $M$ , se triangula la superficie del modelo, en función a la resolución que se desee obtener del mismo.

Cada triángulo  $T$ , constará de un centro  $T_c$ , un vector normal  $T_n$ , y una cámara  $C_i$  que observa de forma fronto-paralela al triángulo, a una distancia determinada, en función a la resolución fijada.

2. Agrupamiento de cámaras.

Una misma cámara puede visualizar a más de un triángulo del modelo.

Para cada cámara  $C_i$ , se define el ángulo  $\lambda$ , el cual es el formado por el rayo principal de visión y el vector normal del triángulo,  $T_n$ .

Para cada triángulo  $T$ , se crea un histograma,  $T_h$ , en el cual se agrupan las cámaras según su ángulo  $\lambda$ , limitando dicho ángulo a 40 grados. Por

lo que para que una cámara vigile correctamente a un triángulo, esta debe estar contenida en un cono recto de visión cuyo eje es el vector normal del triángulo,  $T_n$ , y cuyo ángulo de apertura es de  $80^\circ$ . (Véase Figura 1.42).

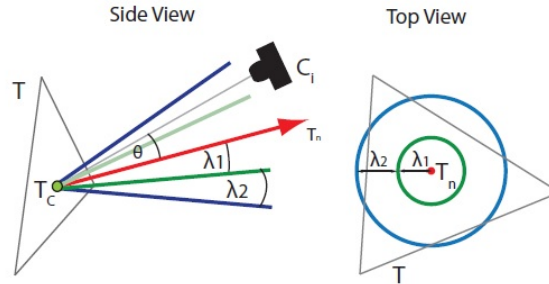


Figura 1.42: Clasificación de las cámaras según el ángulo (Hoppe et al., 2012, p.3).

Cada cámara  $C_i$  tiene asignado un número  $C_iS(C_i)$ , que es el número de triángulos que ve esa cámara, o lo que es lo mismo, el número de histogramas  $T_h$  en los que aparece. Por lo que, cuanto mayor sea ese número, más conveniente será elegir esa cámara.

### 3. Selección de vista.

Los pasos anteriores generan una gran cantidad de cámaras y, por tanto, de información redundante. Por ello, se busca un conjunto de cámaras  $R$  que reconstruya completamente una escena, imponiendo la condición de que debe existir en  $R$  una cámara de cada grupo de cada  $T_h$ .

A continuación se actualizarán los histogramas, eliminando las cámaras que estén ubicadas en el mismo grupo que las cámaras que ya se han escogido anteriormente (Véase Figura 1.43).

### 4. Iteración del algoritmo.

En un ejemplo real, se redujo el número de imágenes a reconstruir de 273 imágenes a 133. Se tomaron 273 imágenes con un MAV, manejado de forma manual. Tras aplicar el algoritmo al modelo, se redujo el número de vistas a 133, usando un MAV de forma autónoma (Véase Figura 1.44).

Por otro lado, se diseñó el camino para que la toma de imágenes se realizara de forma óptima. Comenzando con un grafo completamente conectado, siendo

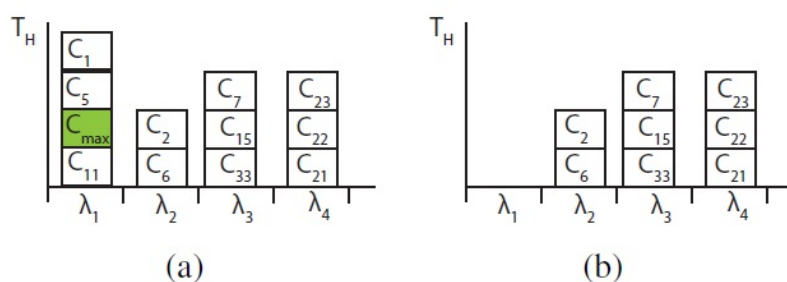


Figura 1.43: Ejemplo de un Histograma de un triángulo T con cuatro grupos. En a) la caja sombreada marca la que ve al mayor número de triángulos. En b) se han eliminado las cámaras del primer grupo tras haber seleccionado la  $C_{max}$  (Hoppe et al., 2012, p.3).

los nodos los centros de las cámaras, se eliminan las aristas que intersectan con el modelo. A cada arista que queda se le asigna un peso, correspondiendo con la longitud de la misma. Para diseñar el camino, se penalizan los cambios en altura y los cambios de dirección. Se comienza por la cámara con el menor peso y se buscará al nodo más cercano (Véase Figura 1.45).

El principal problema que se encuentra es que para mejorar y minimizar las tomas fotográficas se debe realizar una toma de datos previa, y hacer las mejoras sobre un modelo anteriormente obtenido.

De este artículo se sacan las siguientes conclusiones para aplicar al algoritmo que se diseña en la presente tesis:

- Es necesario partir de un modelo previo.
- El ángulo formado entre el vector normal y el rayo principal debe estar limitado.
- Una cámara puede vigilar muchos triángulos, siendo mejor la cámara que más triángulos vigile.
- Es importante diseñar la ruta óptima.
- Este algoritmo se limita a la técnica de la fotogrametría.

#### 1.5.4.3. Algoritmo III

Continuando con el estudio del Diseño de Redes de Fotogrametría, también puede encontrarse el método desarrollado en el artículo «Optimal Camera Net-

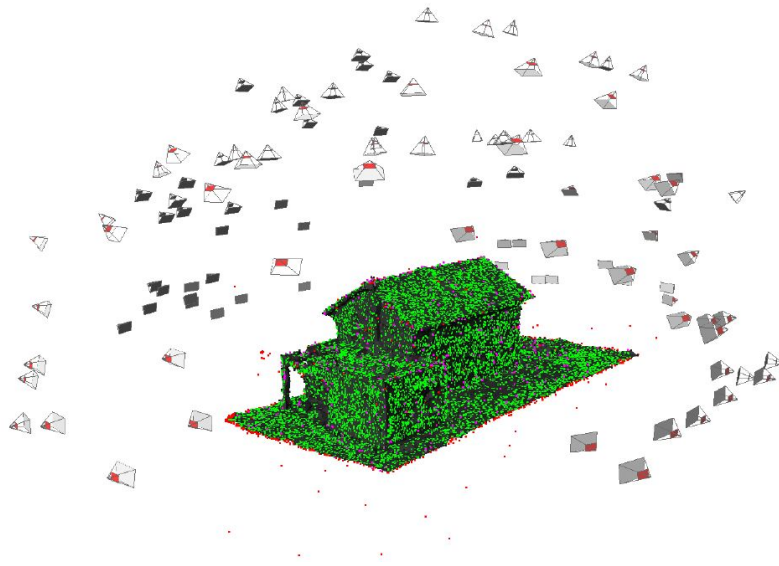


Figura 1.44: Ejemplo de cámaras utilizadas tras aplicar el Algoritmo (Hoppe et al., 2012, p.6).

work Design for 3D Modeling of Cultural Heritage» (Alsadik et al., 2012) para minimizar el número de imágenes para llevar a cabo una reconstrucción tridimensional de un objeto. El método que proponen se basa, a grandes rasgos, en los siguientes pasos:

1. Crear una nube de puntos en bruto.
2. Encontrar el mínimo número de cámaras, filtrando las imágenes iniciales.
3. Optimizar las imágenes filtradas.

Para completar este método se realizarán las operaciones siguientes:

1. Creación de la nube de puntos aproximada.

Con ella se persigue identificar su forma aproximada, su tamaño y su geometría. Para edificios, la nube de puntos se subdivide en pequeñas nubes, cada una correspondiente a una fachada.



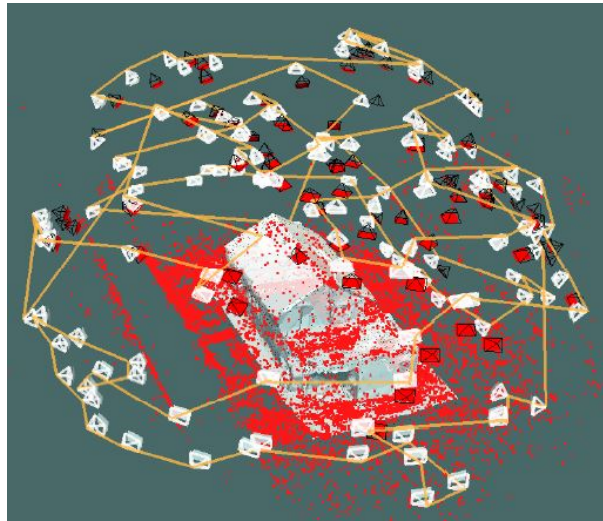


Figura 1.45: Cámaras utilizadas en color gris, cámaras no usadas en color rojo, y camino en color amarillo (Hoppe et al., 2012, p.1).

### 2. Diseño inicial de la red de cámaras.

Se tiene en cuenta el solape que debe existir entre dos imágenes. También el ángulo de inclinación de la toma fotográfica, el cual depende de la altura del edificio. Otro parámetro a tener en cuenta es la distancia entre la cámara y el objeto, y así como el campo de visión de la cámara que se utilice.

La resolución con la que se va a trabajar es un factor clave para determinar la distancia entre el objeto y la cámara.

Se establece así la posición de la cámara  $T_x$ ,  $T_y$  y  $T_z$ , y la orientación de la misma  $\omega$ ,  $\phi$  y  $\kappa$  (ángulos de rotación alrededor de los ejes  $X$ ,  $Y$  y  $Z$  respectivamente).

### 3. Requerimientos de visibilidad.

La nube de puntos, generada automáticamente, se triangula, existiendo para cada triángulo un vector normal. Dependiendo del ángulo que forme el eje óptico de la cámara con el vector normal de un triángulo de la superficie, éste será o no visible. Con este método no se pueden tener en cuenta las áreas que presenten oclusiones.

### 4. Evaluación de la precisión.

Se evalúa la Matriz de Varianza-Covarianza de puntos desconocidos.

5. Minimización del número de cámaras.

A partir de un bloque denso de imágenes, se filtran las cámaras que son redundantes en cuanto a la cubrición del objeto. Tras esto, se modifica y ajusta la posición y la orientación de la cámara para minimizar los errores esperados.

- a) Se divide la nube de puntos en dos zonas: zona sobrecubierta y zona justamente cubierta. Los puntos que están sobrecubiertos son aquellos que aparecen en tres o más cámaras.
- b) El algoritmo comienza en los puntos que presentan sobrecubrición, en orden ascendente, cancelando así primero las cámaras redundantes que están fotografiando a menos puntos.
- c) Se cancelan las cámaras que conllevan a una sobrecubrición de puntos, pero que nunca vigilen también a áreas justamente vigiladas .
- d) La configuración de la red se comprueba de nuevo y se reclasifican las zonas del primer paso.
- e) Se repite el Paso 5c, hasta que no exista ninguna cámara que vigile sólo áreas redundantes.
- f) La configuración de las cámaras filtradas se optimizan para mayor precisión.

Los resultados obtenidos con este método, se pueden resumir con el ejemplo que se ilustra en la Figura 1.46, en el cual se redujo de 31 a 19 el número de cámaras para la reconstrucción.

De este algoritmo se sacan las siguientes conclusiones para aplicar al algoritmo que se diseña en la presente tesis:

- Es necesario partir de un modelo previo.
- El ángulo formado entre el vector normal y el rayo principal debe estar limitado.
- Pueden existir cámaras que vigilen pocos triángulos, pero que sean necesarios para vigilar alguna zona que no es cubierta por ninguna otra

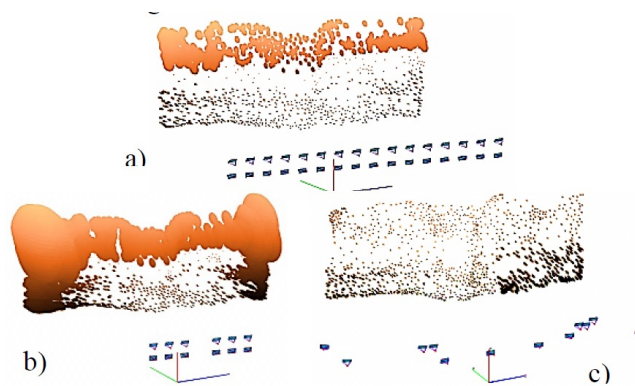


Figura 1.46: a) Red densa, b) Red mínima de cobertura, c) Red mejorada (Alsadik et al., 2012, p.11).

cámara. Por lo que no sólo las cámaras que más triángulos vigilen son las mejores, sino que también son imprescindibles las cámaras que vigilen triángulos o zonas justamente vigilados.

Sin embargo, las carencias que se encuentran a este algoritmo son las siguientes:

- Este algoritmo no tiene en cuenta las oclusiones.
- Este algoritmo se limita a la técnica de la fotogrametría.

### 1.5.4.4. Algoritmo IV

El profesor Ajay Dashora, del Departamento de Ingeniería Civil del Instituto Indio de Tecnología Kanpur, ha centrado su investigación en los últimos años en diseñar algoritmos genéticos que minimicen el tiempo de vuelo para la adquisición de datos a través de tecnología LiDAR<sup>5</sup> (Dashora et al., 2013) (Dashora et al., 2014).

En el que plantea el terreno que debe ser cubierto ha de ser plano u ondulado. Para optimizar este problema se tiene en cuenta el tiempo de vuelo

---

<sup>5</sup>Acrónimo del inglés *Light Detection and Ranging* o *Laser Imaging Detection and Ranging*. Hace referencia a la tecnología que permite determinar la distancia desde un emisor láser a un objeto o superficie utilizando un haz láser pulsado.

requerido para sobrevolar un área de interés dada (AOI), para adquirir datos LiDAR.

Esta operación se realiza trazando cadenas paralelas, entre las cuales, el objeto volador debe posicionarse para empezar la cadena consecutiva. Esta operación aparece ilustrada en la Figura 1.47. Por lo que el tiempo del vuelo tendrá una duración igual al tiempo barrido de la cadena más el tiempo que emplea en dar la vuelta.

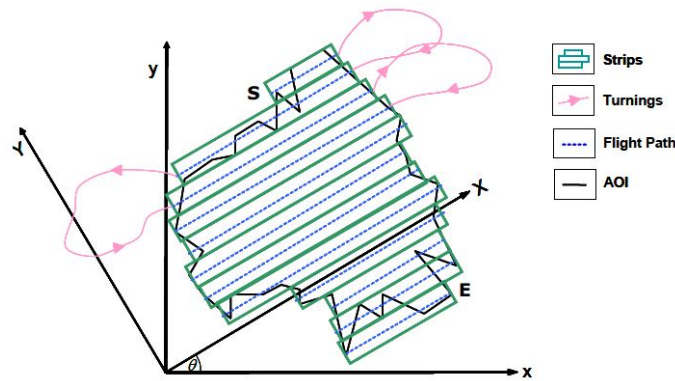


Figura 1.47: Vista esquemática de un AOI, las cadenas de vuelo, y las vueltas (Dashora et al., 2013).

Para ello se diseña un algoritmo genético, en el cual los diferentes parámetros a optimizar son: el ángulo de incidencia,  $\phi$ , la frecuencia del escaneo,  $f$ , la altura del vuelo,  $H$  (o, dicho de otro modo, la distancia entre el instrumento y el objeto), la velocidad,  $V$ , la dirección del vuelo,  $\theta$  y la frecuencia de repetición de pulso,  $PFR$ .

De este algoritmo se sacan las siguientes conclusiones para aplicar al algoritmo que se diseña en la presente tesis:

- El ángulo de incidencia de la toma de datos con respecto al objeto fotografiado es un valor a tener en cuenta.
- La altura del vuelo es importante ya que determina el nivel de detalle de los datos capturados. Este dato se asemeja a la distancia entre el objeto y la cámara.

La limitación que se encuentra en este algoritmo es que se aplica para fotogrametría aérea, en la cual las superficies han de ser onduladas o planas.

### 1.5.4.5. Algoritmo V

En el artículo que tiene por nombre «Viewpoint optimization using genetic algorithm for flying robot inspection of electricity transmission tower equipment» (Yang et al., 2014), se describe el algoritmo genético diseñado con el fin de inspeccionar una torre de transmisión de electricidad, mediante el uso de drones.

La importancia de realizar una buena inspección de estos elementos es fundamental para evitar pérdidas de carga, y así beneficiar tanto al consumidor, como a la empresa.

Tradicionalmente, estas tareas de inspección se han desarrollado a pie, o también, posteriormente, con helicópteros. Ambas presentan problemas, como el tiempo que se consume, y el elevado coste que esto supone. Además, no se garantiza que la inspección se realice de forma completa.

Traduciendo las palabras que aparecen en el citado artículo, se puede encontrar una gran similitud con el problema que existe en el trabajo de toma de datos para un levantamiento arquitectónico: «...Uno de los problemas más importantes que causan este tipo de cuestiones, es que todos los puntos de vista son seleccionados de forma manual, con la experiencia de los inspectores, los cuales, inevitablemente, tienen juicios diferentes» (Yang et al., 2014, pp.426).

En el algoritmo propuesto, por tanto, se trata de dar solución a un problema similar al planteado en la presente tesis: cubrir la vigilancia de un elemento mediante un algoritmo que garantice que todos los puntos son correctamente inspeccionados, con el mínimo esfuerzo posible.

Recientemente se ha introducido el trabajo con robots voladores, los cuales presentan grandes ventajas con respecto a los métodos de inspección tradicionales; pero que, por contra, añaden el problema de contar con un tiempo de batería limitado.

Los factores que influyen en la calidad de una imagen son la posición del punto de vista, así como la orientación.

El algoritmo genético que se ha desarrollado para dar solución a este problema parte de una región alrededor de la torre, en la cual se encontrarán los puntos de vista candidatos (Véase Figura 1.48). Éstos forman una superficie cilíndrica alrededor del objeto.

Las regiones son cuadrículas, y el centro de cada cuadrado será un posible

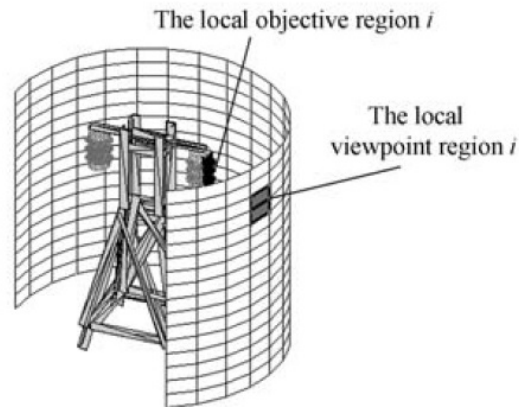


Figura 1.48: Región de búsqueda de puntos de vista (Yang et al., 2014).

punto de vista (Véase Figura 1.49). El elemento a vigilar se representa mediante un modelo triangulado formado por porciones, las cuales reciben el nombre de *patches*.

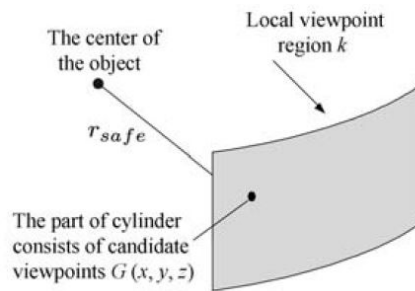


Figura 1.49: Generación de puntos de vista candidatos. La región local del punto de vista  $k$  es una parte de una superficie cilíndrica (Yang et al., 2014).

Siguiendo con la metodología de los algoritmos genéticos, se diseña una función objetivo que controla los siguientes aspectos:

- Visibilidad: porcentaje de triángulos del modelo vigilados.
- Calidad: ángulo que forma el vector normal del triángulo vigilado con el vector de la dirección de la cámara.
- Distancia: ésta tiene que estar comprendida entre una distancia mínima y máxima.

### Función Objetivo del Algoritmo.

La Función Objetivo cuantifica las propiedades de los puntos de vista candidatos. Estos puntos de vista deben ser designados en tanto a que tengan una buena visión de la porción que le corresponda, y que además sea de la mayor calidad posible.

Los factores que se tienen en cuenta para cuantificar la idoneidad de cada uno de esos puntos de vista, son tres: la visibilidad, la calidad de la visión y la distancia.

- Visibilidad ( $f_{vis}$ ). Este factor representa el porcentaje de *patches* que son visualizados con respecto al total de *patches* que componen el objeto.
- Calidad de la visión ( $f_{qua}$ ). Se utiliza para medir la calidad de la imagen percibida. Depende del ángulo formado entre el vector que va del punto de vista al centro del *patch*  $v_i$ , y el vector normal del mismo *patch*,  $n_k$  (Véase Figura 1.50).

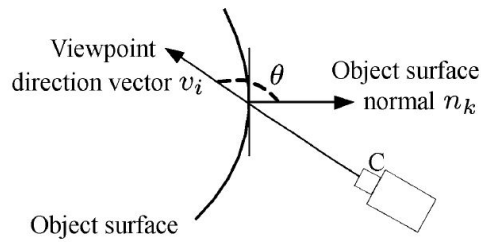


Figura 1.50: Representación del vector normal de la superficie,  $n_k$  y el vector la cámara,  $v_i$  (Yang et al., 2014).

- Distancia ( $f_{dis}$ ). Un mismo *patch* puede ser observado por distintas cámaras. El mejor punto de vista será aquel que vigile al *patch* a una distancia menor.

De esta forma, la función objetivo será la suma de estos tres factores. A cada uno de esos factores se le asigna un peso, en función a los requerimientos que se establezcan.

$$f_{evaluate} = \lambda_{vis}f_{vis} + \lambda_{qua}f_{qua} + \lambda_{dis}f_{dis}$$

En la Figura 1.51 se muestran las seis imágenes para la correcta vigilancia de una pieza. Por cada punto de vista, se puede leer el valor de ajuste obtenido.


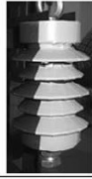
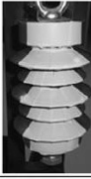

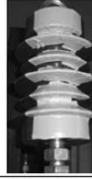
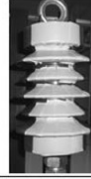
Candidate viewpoint 1	Candidate viewpoint 2	Candidate viewpoint 3	Candidate viewpoint 4	Candidate viewpoint 5	Candidate viewpoint 6
$f_{evaluate} = 0.6316$	$f_{evaluate} = 0.8974$	$f_{evaluate} = 0.9290$	$f_{evaluate} = 0.8915$	$f_{evaluate} = 0.8260$	$f_{evaluate} = 0.9826$
					

Figura 1.51: Imágenes capturadas por puntos de vista candidatos (Yang et al., 2014).

De este algoritmo se sacan las siguientes conclusiones para aplicar al algoritmo que se diseña en la presente tesis:

- La idoneidad de un punto de vista candidato depende tres factores: del porcentaje del objeto que vigila, de la calidad con la que vigila, y de la distancia desde la cual vigila.
- El ángulo formado entre el vector normal y el rayo principal es un factor primordial para evaluar la idoneidad de un punto de vista candidato.
- La distancia del objeto al punto de vista debe estar limitada.
- Se contabiliza el porcentaje de *patches* vigilados.
- La función objetivo diseñada está compuesta por distintos parámetros, cada uno de los cuales con un peso distinto.

Se debe tener en cuenta que en este caso no se trata de realizar una reproducción tridimensional del objeto, solamente de tener controlada toda la pieza. Por tanto no se aplican requerimientos de la fotogrametría como el solape.

#### 1.5.4.6. Algoritmo VI

Actualmente existe una interesante investigación llevada a cabo por la Universidad Julius-Maximilians, en Würzburg, Alemania (Borrmann et al., 2015).

Consiste en la programación de un robot, cuyo nombre es 3Dirma (*Intelligent Robot for Mapping Applications in 3D*). Dicho robot está compuesto por



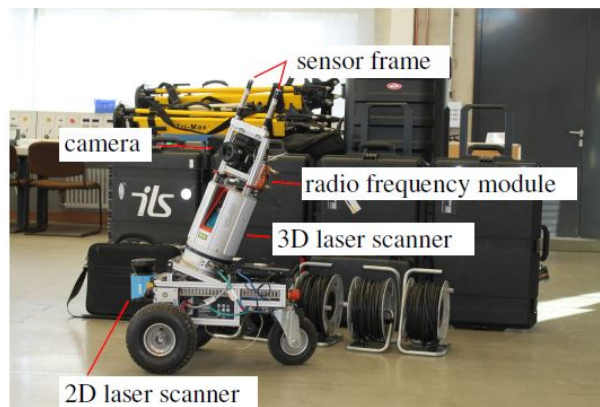


Figura 1.52: Robot 3DIrma (Borrmann et al., 2015).

una cámara de fotos, para aportar la información del color, y un escáner láser sobre un pequeño vehículo de tres ruedas (Véase Figura 1.52).

Para realizar el escaneo de diferentes áreas interconectadas, se requiere un gran número de posiciones. El cambio de posición de estos escaneos es un trabajo que se realiza de forma manual. Para determinar la posición exacta de cada uno de los posicionamientos se ubican dianas especiales que se leen con una estación total. Posteriormente, la posición del escáner es determinada colocando otra diana en el trípode, en la posición en la cual el escáner está. Este proceso requiere un consumo de tiempo, que además se repite para cada posicionamiento nuevo.

La idea de usar un robot para la adquisición de datos del escáner, es eliminar este tiempo y, por tanto, agilizar el proceso de toma de datos.

En el artículo se describe el programa desarrollado y su aplicación a dos escenarios: Ostia Antica (Italia) y el Palacio Würzburg (Alemania).

Se colocan elementos reflectantes que actúan como dianas a lo largo de todo el escenario. Los escaneos se van registrando de forma automática. El programa diseñado une las nubes de puntos mediante una operación de giro. El robot detecta los puntos de vista desde los cuales realizar la toma de datos, reduciendo el tiempo de trabajo en un 75 % (Véase Figura 1.53).

De esta investigación se saca la conclusión de que es necesario controlar la planificación de los estacionamientos para reducir el tiempo de la adquisición de datos.

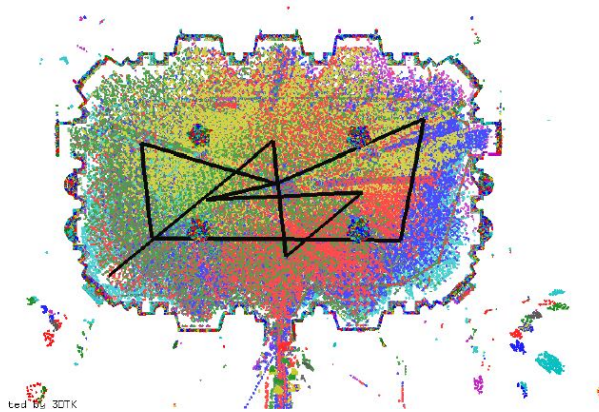


Figura 1.53: Planta escaneada del Salón Imperial del Palacio Würzburg y la ruta realizada (Borrmann et al., 2015).

#### 1.5.4.7. Análisis de Visibilidad con QGIS

El *software QGIS* es un Sistema de Información Geográfica (SIG)<sup>6</sup>, de código libre, el cual permite crear, editar, visualizar y analizar información geoespacial a través de formatos *ráster* y vectoriales, así como bases de datos. La versión actual es QGIS 2.18.6 «Las Palmas» (QGIS, 2017).

El *software* es desarrollado por distintos voluntarios y organizaciones, los cuales crean módulos o extensiones que añaden alguna función al programa.

Pues bien, la extensión *Viewshed Analysis* permite realizar avanzados análisis de visibilidad. Esta extensión fue incorporada por primera vez en Enero de 2014. Es capaz de calcular qué áreas pueden ser vistas desde uno o más puntos dados, o, al contrario, desde qué áreas puede ser visto un punto.

Será necesario determinar la porción de terreno con información altimétrica del mismo, la altura del observador y la distancia máxima de observación, como puede observarse en la Figura 1.54.

Esta aplicación resulta muy útil en trabajos medioambientales y para la evaluación del impacto ambiental, o percepción del paisaje. Un ejemplo de aplicación es el mostrado en la Figura 1.55, en la que se establece una altura del observador de 1,6 metros, y un ángulo de visión de 360°, así como una distancia máxima de visión de 500 metros.

<sup>6</sup>También se puede encontrar como GIS, *Geographic Information System*

## 1.5. ESTADO DE LA CUESTIÓN

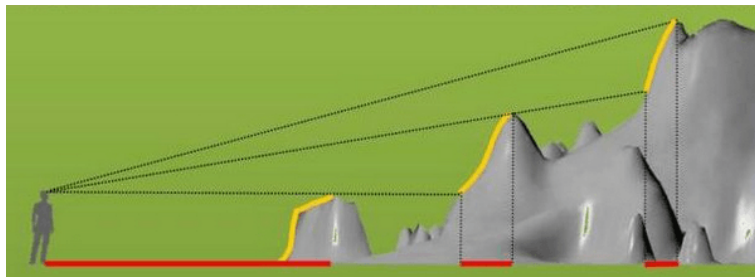


Figura 1.54: Principio de funcionamiento del complemento *Viewshed Analysis* de QGIS (Alonso, 2016).

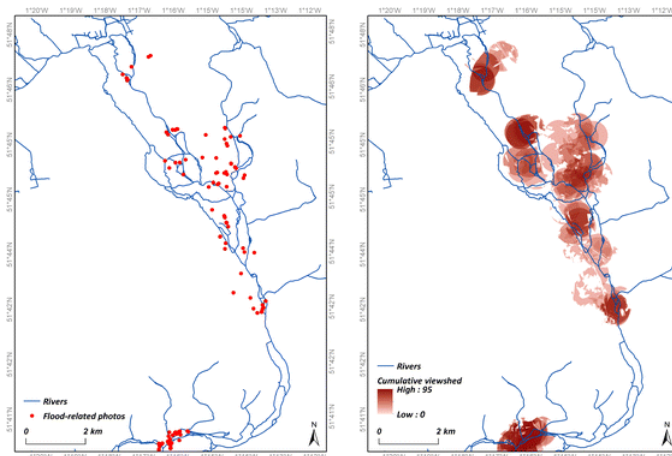


Figura 1.55: Aplicación de la herramienta *Viewshed Analysis* de QGIS (Rosser et al., 2017).

Además, permite generar redes de intervisibilidad, es decir, es capaz de conectar redes de relaciones visuales entre conjuntos de puntos.

De esta aplicación es interesante el uso de la altura del observador, así como de la distancia máxima hasta la cual es posible observar de forma adecuada. También se debe mencionar la posibilidad que ofrece de crear redes de puntos de vista que se visualizan entre ellos. Este principio está muy conectado con los de la Topografía, por el cual se necesita de un grafo interconectado de posiciones para poder reconstruir la toma de datos desde distintos puntos de vista.

Sin embargo, este método resulta útil siempre y cuando se tenga información tridimensional del objeto a vigilar, lo cual no siempre ocurre. Pueden existir edificios, yacimientos, ruinas u otros elementos, los cuales no se en-

cuentren correctamente digitalizados en cuanto a su altimetría. Por otro lado, la elección de los puntos de vista desde los cuales realizar la toma de datos se hace de forma manual e intuitiva, sin aplicar algún método de búsqueda de los mejores puntos de vista.

## Capítulo 2

# Objetivos

El objetivo principal de este trabajo de investigación es diseñar un novedoso protocolo de actuación por el cual se establezcan los puntos de vista desde los cuales ejecutar la toma de datos de un edificio para la realización del levantamiento arquitectónico del mismo. Dicho protocolo servirá para dotar a los profesionales de una herramienta que por una parte les asista a la hora de planificar los posicionamientos para un levantamiento arquitectónico y, por otra, les permita comprobar con algún margen de seguridad que la toma de datos ha sido suficiente.

Para ello, los objetivos específicos que se persiguen son los siguientes:

- Analizar las distintas técnicas e instrumentos de topografía existentes para la realización de levantamientos arquitectónicos.
- Analizar las distintas técnicas matemáticas de optimización aplicables al problema de vigilancia de un objeto.
- Adaptar dichas técnicas matemáticas de optimización a los fundamentos de la topografía y a las restricciones impuestas por el instrumento a utilizar, diseñando un método capaz de optimizar el proceso de toma de datos.
- Seleccionar el lenguaje de programación más adecuado para la implementación del método diseñado.
- Aplicar dicho método en caso de estudios concretos, realizando el levantamiento arquitectónico de edificios, aplicando las técnicas más adecuadas para la adquisición y gestión de los datos.



## Capítulo 3

# Metodología

El programa de doctorado en el cual se encuentra inscrito el proyecto de esta tesis es el denominado **Tecnología de la Construcción, Investigación, Desarrollo e Innovación**, de la Escuela Técnica Superior de Ingeniería de Edificación de la Universidad de Sevilla.

Dado que este proyecto de tesis relaciona dos áreas de conocimiento diferentes, se ha trabajado de forma conjunta tanto con el **Departamento de Ingeniería Gráfica**, en el área de la Expresión Gráfica Arquitectónica, como con el **Departamento de Matemática Aplicada I**, ambos de la **Universidad de Sevilla**. También ha colaborado la Escuela Técnica Superior de Informática de Sevilla, y en concreto, el **Departamento de Ciencias de la Computación e Inteligencia Artificial** de esta misma universidad.

El proceso de investigación se ha desarrollado entre los años 2012 a 2017 de la forma en la que a continuación se detalla.

### Estudios Previos.

La línea de investigación que sigue este trabajo nació con el Trabajo Fin de Máster realizado en el año 2010, denominado *Optimización en el Posicionamiento de Estaciones Topográficas en los Levantamientos de Edificios* (Cabrera Revuelta et al., 2010). En él se realizó una revisión bibliográfica de los llamados **Problemas de la Galería de Arte**, así como de las técnicas e instrumentos que se utilizan para la realización de un levantamiento arquitectónico. Además, se propuso un método para optimizar los posicionamientos desde los cuales cubrir la vigilancia de un polígono, basado en el **Problema de la Prisión**.

Los resultados obtenidos fueron expuestos en el VII Encuentro Andaluz de Matemática Discreta, celebrado en Carmona, en Octubre de 2011 (Cabrera Revuelta et al., 2011). Fue en ese evento donde se sugirió centrar la investigación en otro tipo de técnica de optimización, proponiéndose las llamadas técnicas metaheurísticas. A partir de ese momento comenzó el trabajo de investigación de la presente tesis doctoral.

Cabe mencionar que al ser ésta una línea de investigación que no continúa trabajos previos de investigadores del mismo área, se ha necesitado invertir tiempo en encuadrar el tema y encontrar las palabras claves utilizadas para la búsqueda de trabajos similares, por lo que éste puede considerarse como una trabajo de investigación de exploración.

### Proceso de investigación.

El trabajo de investigación se ha desarrollado de dos formas, llevadas a cabo en paralelo. Por un lado se ha realizado una búsqueda de bibliografía relacionada, y a su vez se han ido diseñando los métodos que se presentan. Estas fases se han intercalado.

#### ■ Revisión Bibliográfica.

En primer lugar, el trabajo de investigación comenzó con una extensa revisión bibliográfica en dos vertientes: por una lado, la relacionada con las técnicas de levantamiento arquitectónico, con el fin de conocer las técnicas e instrumentos que se utilizan en la actualidad, así como para conocer las distintas recomendaciones existentes para la planificación de un trabajo de levantamiento arquitectónico; y, por otro, se realizó un análisis más profundo sobre los Problemas de la Galería de Arte.

Más adelante, se emprendió el estudio relativo a las Técnicas de Optimización Metaheurísticas, prestando especial interés a los **Algoritmos Genéticos**. Para ello se consultaron libros, tesis doctorales, así como artículos académicos.

Por otro lado, se inició una nueva búsqueda bibliográfica orientada a conocer métodos de optimización en redes de fotogrametría. Gracias a la colaboración del Profesor Diego González Aguilera, Catedrático en el Área de Ingeniería Cartográfica, Geodésica y Fotogrametría en la Universidad de Salamanca, la revisión se centró alrededor de los Problemas de Diseño de Redes de Fotogrametría (PND-*Photogrammetric Network Design*). Aunque el problema planteado en este trabajo de investigación no es exactamente el PND, ambos se encuentran estrechamente relacionados.



---

A partir de este momento se abrió un amplio campo de búsqueda y fueron localizados multitud de artículos en los que se trataban temas similares al objeto de estudio de esta investigación (Irschara et al., 2010), (Alsadik et al., 2012), (Yang et al., 2014). Muchos de estos artículos han servido de inspiración para diseñar los métodos que aquí se presentan, tomando ideas como que la distancia del objeto al instrumento, así como el ángulo de incidencia son dos parámetros necesariamente ineludibles.

■ Diseño de los métodos

• Diseño del primer método.

Una vez estudiados los Problemas de la Galería de Arte, y prestando especial interés al **Problema de la Fortaleza**, se comenzó con el diseño del primer método de optimización en el posicionamiento que se presenta en este trabajo.

En esta fase se desarrolla un estudio teórico previo que establece el procedimiento para la formación de áreas de visibilidad suficientes desde las cuales cubrir la vigilancia externa de un edificio, el cual es representado por el polígono que conforma su vista en planta. El método diseñado se trata de un algoritmo determinista, pues la solución que se obtiene para unos datos de entrada concretos, es siempre la misma.

Los resultados obtenidos fueron presentados en el XI Congreso Internacional de Expresión Gráfica Aplicada a la Edificación, celebrado en Valencia, en Noviembre de 2012 (Cabrera Revuelta et al., 2012).

• Diseño del segundo método.

Tras un estudio detallado de los fundamentos de los Algoritmos Genéticos, se realiza una primera aproximación del método basado en técnicas metaheurísticas. En ella se tienen en cuenta las áreas de visibilidad obtenidas a través del primer método diseñado.

Este segundo método diseñado, a diferencia del primero, es un algoritmo no determinista. Esto quiere decir que para unos datos de entrada, pueden obtenerse distintas soluciones.

Esta primera aproximación es un estudio teórico previo, para el cual se necesitó el apoyo del *software* AUTOCAD para la realización de las simulaciones necesarias.

Los resultados de esta primera aproximación fueron expuestos en el IX Encuentro Andaluz de Matemática Discreta, celebrado en Almería, en Octubre de 2015 (Cabrera Revuelta et al., 2015b).

El siguiente paso, como no podía ser de otra manera, fue la implementación de este algoritmo. Es sabido que los algoritmos genéticos no tienen sentido si no se encuentran implementados, ya que su funcionamiento se basa en la selección de datos aleatorios, y de operaciones de cruce también aleatorias siendo necesario manejar y ordenar una gran cantidad de datos. El proceso se repite un número elevado de iteraciones, siendo necesario calcular el Valor de Ajuste para cada posible solución.

### Proceso de implementación.

#### ■ Primer método.

El método de optimización en el posicionamiento para la realización de levantamiento arquitectónicos, basado en la el Problema de la Fortaleza fue propuesto al Proyecto Sinergia 2014-2015. Dicho proyecto promueve el fomento del trabajo cooperativo multidisciplinar y el espíritu de emprendimiento mediante Trabajos Fin de Grado y Trabajos Fin de Máster.

De esta forma, este tema fue elegido en la Escuela Técnica Superior de Informática de Sevilla como Trabajo Fin de Grado, para su implementación. Esto dio como resultado el Trabajo Fin de Grado denominado *Aplicación web de posicionamiento topográfico* (Martín López, 2015b), dirigido por el Profesor Doctor Jose Ramón Portillo. Se realizó una implementación en PYTHON del método, el cual tiene como entrada las coordenadas XY que representan los vértices de un edificio, y el orden de unión entre ellos; y, como datos de salida, se obtienen las áreas de visibilidad suficientes para vigilar el contorno del edificio.

Este trabajo fue premiado en la II Edición de los Premios COORDINA de la Universidad de Sevilla <sup>1</sup>

#### ■ Segundo método.

En un primer momento se pensó en su implementación a través del complemento *Galapagos*. *Galapagos* es el complemento de la aplicación *Grasshopper*, que a su vez es un complemento de *Rhino* y que permite utilizar algoritmos genéticos para resolver problemas de optimización.

El problema denominado «Algoritmo Genético para el Diseño de Redes de Fotogrametría Terrestre» fue propuesto como caso de estudio a las **Jornadas FORMA 15**. Estas jornadas son organizadas conjuntamente

---

<sup>1</sup>Mejor Proyecto relacionado con la introducción de Nuevas Tecnologías en la Edificación.

---

por el Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Sevilla y la Asociación Cultura Digital (CulturaDigital, 2015). Conforman una iniciativa que busca acercar algunas herramientas de simulación a colectivos de múltiples ámbitos (Ciencias Sociales, Biología o Economía, entre otros) con el fin de profundizar en el conocimiento de los procesos complejos.

A partir de estas jornadas se creó un equipo multidisciplinar junto con informáticos, y desde entonces se ha trabajado en mejorar los resultados que se obtuvieron en ellas. Para la implementación del algoritmo genético, se ha utilizado el lenguaje *Net-Logo*, basando el modelo en un Sistema Multiagente.

Los últimos resultados obtenidos fueron expuestos en el XIII Congreso Internacional de Expresión Gráfica aplicada a la Edificación, celebrado en Castellón de la Plana, en diciembre de 2016 (Cabrera Revuelta et al., 2017).

En la actualidad se está realizando un Trabajo Fin de Grado en Tecnologías Informáticas, con el fin de implementar el algoritmo genético diseñado en lenguaje JAVA.

### **Proceso de aplicación y obtención de resultados.**

Como caso de uso se ha elegido el Molino de Marea de Río Arillo por ser un edificio emblemático situado en la Bahía de Cádiz, y por contribuir a su puesta en valor, dado el estado ruinoso en el que se encuentra dicho inmueble. Además, el edificio ha resultado idóneo al ser de poca altura, con una planta lo suficientemente compleja como para ser considerada la importancia de una buena planificación para la ejecución del levantamiento. A esto hay que añadir el entorno en el cual se encuentra el molino, ya que a su alrededor hay zonas fangosas, carretera y el caño del Río Arillo. Toda esta dificultad hace que el campo de búsqueda en el cual posicionar la estación o escáner sea limitado.

Los aparatos utilizados para la realización del levantamiento han sido el escáner láser Leica C10 y la cámara panorámica iSTAR 360°. Para elegir el día de ejecución del trabajo de campo hubo que prestar atención a la tabla de mareas, ya que era conveniente que ésta se encontrara lo más baja posible. Por tanto, se localizó un día con un coeficiente de marea alto, en el que la bajamar coincidiera con una buena hora de sol.

Aplicando el modelo del Algoritmo Genético diseñado, por medio de un Sistema Multiagente desarrollado en *Net-Logo*, se ha obtenido un conjunto de

posicionamientos desde los cuales se cubría la vigilancia desde el exterior del edificio.

Los datos obtenidos se han procesado con el *software Cyclone* y se han georeferenciado. Como resultado se ha obtenido la nube de puntos completa del edificio, por lo que se pudo comprobar que el algoritmo genético ha sido eficiente.

Los datos obtenidos servirán como base para futuros trabajos de investigación acerca del edificio, como Trabajo Fin de Grado en Edificación.

#### **Trabajos complementarios.**

Para complementar el trabajo, se han realizado también pruebas encaminadas a determinar cuál es el ángulo mínimo que debe existir entre el rayo láser emitido por el escáner y la superficie escaneada, a partir del cual la toma de datos se puede considerar lo suficientemente buena. Para ello se escanearon diferentes paramentos y se marcaron dianas sobre los mismos, formando ángulos determinados con el rayo láser. Posteriormente se ha analizado el error cometido por el instrumento, en función al ángulo de incidencia.

## Capítulo 4

# Marco Teórico

En este capítulo se procede a explicar con detalle los conceptos matemáticos que se han estudiado y que han sido necesarios para desarrollar los algoritmos diseñados.

Por ello, este capítulo está dividido en dos partes diferenciadas:

- Una primera parte dedicada a los **Problemas de Visibilidad**, haciendo un recorrido por las distintas variantes del mismo.
- Una segunda parte centrada en los elementos necesarios para diseñar un **Algoritmo Genético**.

## 4.1. Revisión bibliográfica sobre los Problemas de Visibilidad

Dentro de la **Geometría Computacional**, los **Problemas de Visibilidad** han suscitado un gran interés a lo largo de la Historia a la comunidad científica. En esta sección se realiza una revisión bibliográfica sobre los mismos.

Los Problemas de Visibilidad nacen con el llamado **Problema de la Galería de Arte**, planteado por Victor Klee en 1973 (Danaraj and Klee, 1974). Klee pensó que las obras de arte expuestas en las galerías deben ser correctamente vigiladas debido a su incalculable valor. Para reducir los costes de vigilancia, interesa que el número de guardias que vigilen toda la superficie de dicha galería sea el mínimo posible.

Por lo que el problema fue planteado de la siguiente forma: Dada una sala de un museo con  $n$ -muros, ¿cuántos guardias son necesarios para vigilar de forma completa el interior de esta sala de  $n$ -muros?

En esta disciplina se combinan la geometría, la computación y la combinatoria, creando resultados de gran aplicación en campos como la Robótica, la Planificación de Trayectorias, la Visión por Ordenador y la Arquitectura Asistida por Ordenador.

Desde que se planteara el Problema de la Galería de Arte, numerosos matemáticos han centrado su trabajo en este tipo de estudios de vigilancia, desarrollando distintas variantes del mismo.

El profesor Joseph O'Rourke ha sido uno de los primeros investigadores más reconocidos en este campo. Su libro *The Art Gallery Theorems and Algorithms* (O'Rourke, 1987) es una investigación monográfica sobre el **Problema de la Galería de Arte** y todas sus variantes estudiadas entre los años 70 y 80. Tal y como él describe, esta materia es fascinante y accesible, y en su opinión debería estar disponible para una audiencia más amplia.

Por otro lado, el profesor Jorge Urrutia, Profesor de la Universidad Nacional Autónoma de México, y cuya área principal de trabajo es la Geometría Discreta y Computacional, es uno de los investigadores más relevantes de estos últimos años sobre los Problemas de la Galería de Arte, habiendo dirigido numerosas tesis doctorales en esta materia. En su libro *Art Gallery and Illumination Problems* (Urrutia et al., 2000), escrito diez años después del mencionado en el párrafo anterior, plantea nuevos problemas y resuelve algunos planteados por el profesor O'Rourke.

En la Tesis Doctoral de Santiago Canales, investigador de la Universidad Politécnica de Madrid, denominada *Métodos Heurísticos en Problemas Geométricos. Visibilidad, Iluminación y Vigilancia* (Canales Cano, 2004), se realiza un análisis de cada una de las variantes del Problema de la Galería de Arte.

Más recientemente se ha realizado un estudio monográfico sobre los problemas de la galería de arte más actuales, describiendo los desarrollos obtenidos de forma cronológica, y comparando algoritmos actuales (de Rezende et al., 2016).

Es importante no perder de vista que en muchas ocasiones, los elementos físicos reales discrepan de los modelos teóricos utilizados, con lo cual los resultados obtenidos no son directamente aplicables.

Para comenzar, es necesario definir el concepto de **polígono**: «Un polígono  $P$  es una secuencia ordenada de  $n$  puntos  $v_1, v_2, \dots, v_n$ , con  $n \geq 3$ , llamados vértices de  $P$ , junto con el conjunto de líneas uniendo  $v_i$  a  $v_{i+1}$  para  $i = 1, 2, \dots, n-1$  y  $v_n$  a  $v_1$  llamadas aristas de  $P$ » (Morales, 2006). Un Polígono  $P$  es, por tanto, una colección de  $n$  vértices y  $n$  aristas, tal que ningún par de aristas no consecutivas compartan un punto. El conjunto de vértices y de aristas será referido como el *Borde de  $P$* , denotado por  $\delta P$ . El borde de un polígono es una *Curva Jordan*: separa el plano en dos regiones disjuntas, el interior y el exterior del polígono.

La idea de **Visibilidad** mantiene que, dado un conjunto  $D$  en  $\mathbb{R}^2$  se dice que los puntos  $x, y \in D$  son visibles en  $D$  cuando el segmento  $xy$  está completamente contenido en  $D$ . El conjunto  $D$  se llamará *convexo* si cualquier par de puntos del mismo se ven. Un conjunto se dice *estrellado* si existe un punto que ve a todos los demás.

##### 4.1.1. El Problema de la Galería de Arte

En 1973 profesor Victor Klee planteó el problema de determinar el mínimo número de guardias suficiente para cubrir el interior de una habitación de  $n$ -muros.

El profesor Václav Chvátal dos años después dio respuesta a este problema con el célebre Teorema de la Galería de Arte 4.1.1, el cual establece que  $\lfloor n/3 \rfloor$  guardias son ocasionalmente necesarios y siempre suficientes para cubrir un polígono de  $n$  vértices (Chvatal, 1975).

**Teorema 4.1.1 (Teorema de la Galería de Arte. Chvátal, 1975)**  $\lfloor n/3 \rfloor$  guardias son ocasionalmente necesarios y siempre suficientes para cubrir un polígono de  $n$  vértices, (Chvatal, 1975).

Por lo tanto,  $\lfloor n/3 \rfloor$  luces o guardias son siempre suficientes para iluminar o vigilar un polígono  $P$ , de  $n$  vértices, pero en muchas ocasiones basta con menos. Por lo que se plantea el problema algorítmico siguiente: dado un polígono, calcular el mínimo número de luces que lo vigilan. Este problema fue estudiado por Lee y Lin (Lee and Lin, 1986), probando que es un problema de naturaleza  $NP$ -completo, por lo que es preferible encontrar el *número suficiente y ocasionalmente necesario de guardias* de tal forma que  $P$  quede vigilado.

Se dice que un punto  $x \in P$  *ve o cubre* un punto  $y \in P$  si el segmento de línea  $xy$  es un subconjunto de  $P$ .

Para cualquier polígono  $P$ , se define  $G(P)$  como el menor número de puntos de  $P$  que cubre a todo  $P$ . Se define  $g(n)$  como el máximo valor de  $G(P)$  sobre todos los polígonos de  $n$  vértices, es decir, representa el máximo número de guardias que son siempre necesarios para un  $n$ -polígono. Así, el valor  $g(n)$  guardias son ocasionalmente necesarios y siempre suficientes, o simplemente necesarios y suficientes.

#### 4.1.1.1. Necesidad

Con formas simples como el triángulo, se observa que  $g(3) = 1$ , pero, a medida que la forma poligonal aumenta su complejidad, no es fácil detectar el valor de  $g(n)$ . El caso del peine es un ejemplo de ello. En él, cada púa necesita su propio guardia. El peine de  $k$  púas, tendrá  $n = 3k$  vértices, por tanto, se establece que  $g(n) \geq \lfloor n/3 \rfloor$  (Véase Figura 4.1).

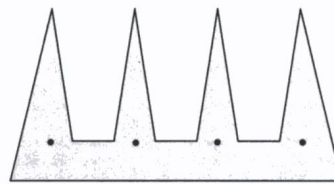


Figura 4.1: El peine de Chvátal (O'Rourke, 1998, p.6).

Se debe diferenciar entre dos tipos de guardias: los **Guardias en Vértices**, sólo pueden estar ubicados en los vértices del polígono; y los **Guardias en**



**Puntos**, los cuales no tienen restricción en su ubicación. Se define  $g_v(n)$  como el número de vértices necesarios y suficientes para cubrir un polígono  $n$  lados. En algunos polígonos  $g_v(n) \neq g(n)$  (hay polígonos que necesitan varios guardias en vértices, pero con uno sólo en su interior sería suficiente), pero, ya que  $g(n)$  resume la información sobre todos los polígonos, se reduce a que  $g_v(n) = g(n)$ , simplificando las soluciones a los guardias en vértices.

##### 4.1.1.2. Prueba de Fisk

Steve Fisk prueba el Teorema 4.1.1 de Chvátal de una forma sencilla y elegante, (Fisk, 1978): Obténgase una triangulación arbitraria del polígono, 3-coloréese el grafo de dicha triangulación y colóquense luces en los vértices que van etiquetados con el color menos frecuente de la coloración.

El primer paso en la **Prueba de Fisk** es triangular el polígono  $P$  añadiendo diagonales internas entre vértices.

En este momento es necesario mencionar que cualquier polígono  $P$  con  $n$  vértices admite una triangulación, y cualquier triangulación consta exactamente de  $n - 2$  triángulos, mediante la unión de  $n - 3$  diagonales internas.

El segundo paso es recordar que el grafo de un polígono triangulado puede ser 3-coloreado. Una  $k$ -coloración de un grafo consiste en asignar colores a los nodos, usando no más de  $k$  colores, de forma que no haya dos nodos adyacentes con el mismo color. Ya que un grafo de triangulación es plano, éste es 4-coloreable, por el conocido **Teorema de los Cuatro Colores** 4.1.2, pero al tratarse de un polígono triangulado, se puede reducir a que su grafo es 3-coloreable (Véase Figura 4.2).

##### **Teorema 4.1.2 (Teorema de los Cuatro Colores. Appel y Haken, 1976)**

*Cualquier mapa puede ser coloreado solamente con cuatro colores distintos de tal manera que dos regiones adyacentes no tengan el mismo color; (Appel et al., 1977)*

El tercer paso es notar que uno de los tres colores no será usado más de  $n/3$  de veces.

El cuarto paso es ubicar a los guardias en cada nodo que posea el color menos repetido, ya que cada triángulo poseerá un vértice de ese color, y la suma de todos los triángulos es el interior del polígono.

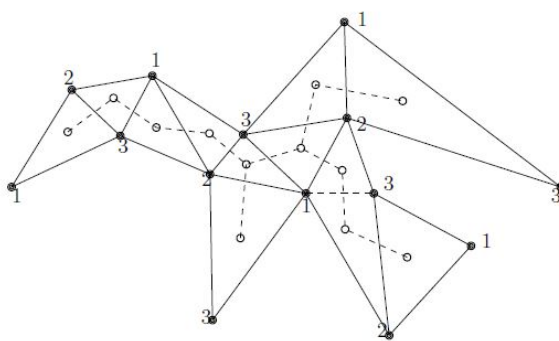


Figura 4.2: Polígono triangulado, 3-coloreado y su gráfica dual (Morales, 2006).

Esto establece que  $\lfloor n/3 \rfloor$  guardias en vértices son suficientes para cubrir el interior de un polígono arbitrario.

#### 4.1.1.3. El Algoritmo de Avis y Toussaint

El Algoritmo desarrollado en el artículo «An Optimal Algorithm for Determining the Visibility of a Polygon from an Edge» por David Avis y Godfried T. Toussaint (Avis and Toussaint, 1981) sigue los pasos de la prueba de Fisk:

1. Triangular  $P$ , obteniendo un grafo  $G$ . Orden computacional  $O(n \log \log n)$ .
2. Tri-colorear los vértices de  $G$ . Orden computacional  $O(n)$ .
3. Colocar los guardias en los vértices con el color menos usado. Orden computacional  $O(n)$ .

El resultado del algoritmo se muestra en la Figura 4.3, y es de orden  $O(n \log n)$ .

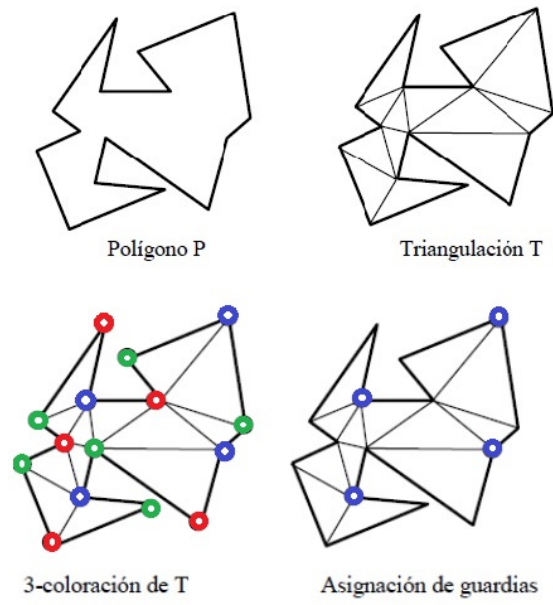


Figura 4.3: Algoritmo de Avis y Toussaint (Morales, 2006).

### 4.1.2. Variantes del Problema de la Galería de Arte

Partiendo del problema más sencillo planteado, el cual trata de vigilar el interior de una galería de arte, diferentes matemáticos han propuesto variantes del mismo. Por ejemplo, ¿cuántos guardias son necesarios para vigilar de forma completa una galería con  $h$  obstáculos en su interior?

A continuación se exponen las diferentes casuísticas desarrolladas a partir del Problema de la Galería de Arte original.

#### 4.1.2.1. Vigilancia de Polígonos con Obstáculos

Una de las distintas variantes del Problema de la Galería de Arte trata de la vigilancia interna de un polígono cuando en su interior existen obstáculos, como pueden ser pilares, que impidan la visión a través de ellos, y que interesa vigilar también. A estos obstáculos se les denomina *hoyos* (en inglés *holes*), y son designados con la letra  $h$ .

**Teorema 4.1.3 (Hoffmann, Kaufmann y Kriegel, 1991)** *Cualquier polígono  $P$  con  $n$  vértices y  $h$  obstáculos puede vigilarse con a lo más  $\lfloor \frac{n+h}{3} \rfloor$  guardias en puntos, (Hoffmann et al., 1991).*

I. Bjorling-Sachs y D.L. Souvaine desarrollan un algoritmo para la vigilancia de este tipo de polígonos (Bjorling-Sachs and Souvaine, 1995).

Con dicho algoritmo demuestran que toda galería de  $n$  vértices y  $h$  obstáculos es vigilada con  $\lfloor (n+h)/3 \rfloor$  guardias. Para algunas galerías este número de guardias es necesario.

En esta variante del Problema de la Galería de Arte, cada obstáculo es conectado con el exterior del polígono a través de un canal (Véase Figura 4.4). Estos canales tienen forma triangular, por lo que por cada obstáculo se añade un nuevo vértice. De esta forma se obtendrá un nuevo polígono parecido al original con  $n + h$  vértices.

Este nuevo polígono puede ser triangulado y de esta forma se aplicaría el algoritmo expuesto en el Apartado 4.1.1.3 (Véase Figura 4.5).

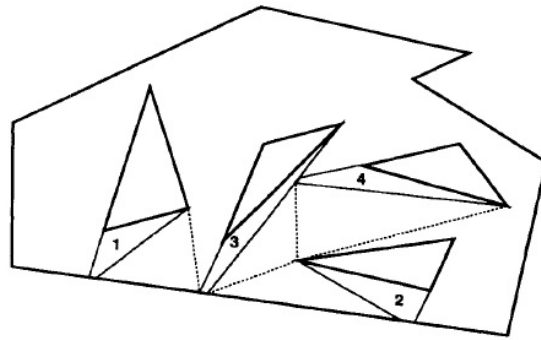


Figura 4.4: Construcción de los canales (Bjorling-Sachs and Souvaine, 1995).

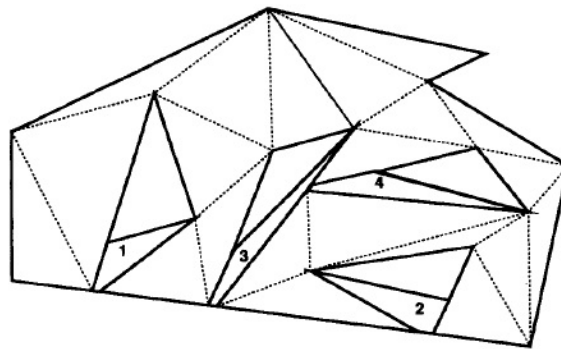


Figura 4.5: Triangulación del nuevo polígono con  $n + h$  vértices (Bjorling-Sachs and Souvaine, 1995).

##### 4.1.2.2. Vigilancia de Polígonos Ortogonales

Un **Polígono Ortogonal** es aquel cuyas aristas están todas alineadas con un par de ejes coordenados ortogonales, los cuales pueden ser horizontal y vertical, sin pérdida de generalidad.

Los ángulos internos formarán  $90^\circ$  o  $270^\circ$ . Son útiles para aproximaciones a polígonos.

La primera vez que se formula el Problema de la Galería de Arte para Polígonos Ortogonales es gracias a Kahn, Klawe y Kleitman en 1983 (Kahn et al., 1983). Establecen que  $\lfloor n/4 \rfloor$  guardias son ocasionalmente necesarios y siempre suficientes para ver el interior de una galería de arte de forma ortogonal con  $n$  vértices, reduciéndose así el número de guardias suficientes para vigilar

un edificio si la planta es ortogonal (Véase Figura 4.6).

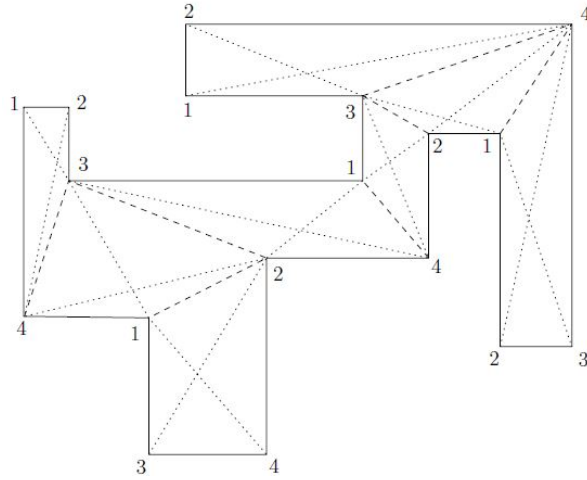


Figura 4.6: Polígono ortogonal cuadrilaterizado, su gráfica  $H$  y 4 coloreado. (Morales, 2006).

**Teorema 4.1.4 (El Problema de la Galería de Arte Ortogonal. Kahn, 1980)**  $\lfloor \frac{n}{4} \rfloor$  guardias son a veces necesarios y siempre suficientes para vigilar el interior de un polígono ortogonal de  $n$  vértices (Kahn et al., 1983).

**La Cuadrilaterización Convexa.** Un polígono ortogonal puede ser dividido en cuadriláteros convexos, de esta forma, al colocar un guardia en un vértice de un cuadrilátero, se vigilará el cuadrilátero completo. En general, la cuadrilaterización convexa no es única, hay diferentes opciones. Es necesario prestar atención a que no basta con unir pares de triángulos, ya que pueden obtenerse cuadriláteros no convexos, como puede observarse en la Figura 4.7.

Es fácil intuir que también se ha desarrollado el problema de vigilar un polígono ortogonal con obstáculos, como puede verse en la Figura 4.8.

**Teorema 4.1.5 (Hoffmann y Kriegel, 1996)** Cualquier polígono ortogonal  $P$  con  $n$  vértices y  $h$  obstáculos puede vigilarse con a lo más  $\lfloor \frac{n}{3} \rfloor$  guardias en vértices, (Hoffmann and Kriegel, 1996).

Por último, recientemente se ha estudiado el problema de vigilar galerías de arte ortogonales, con guardias que se desplazan en una línea, dicho de otro modo, con guardias móviles (Biedl et al., 2017).

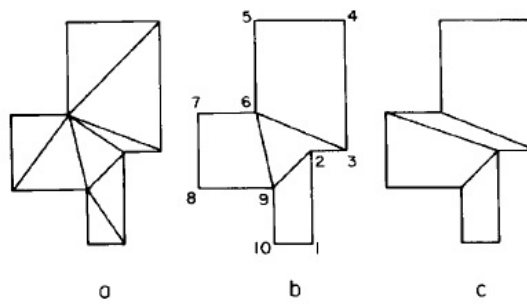


Figura 4.7: a) Triangulación de polígono ortogonal; b) Unión de pares de triángulos no dan como resultados cuadriláteros convexos; c) Correcta cuadrilaterización convexa (O'Rourke, 1987).

**Teorema 4.1.6** *Dado un polígono ortogonal  $P$  con  $n$  vértices, que tiene una segmentación en caminos,  $\lfloor \frac{n+2}{6} \rfloor$  guardias móviles son siempre suficientes y a veces necesarios para vigilar  $P$  de forma completa, (Biedl et al., 2017).*

#### 4.1.2.3. Problema de Visibilidad Exterior

Derick Wood y Joseph Malkelvitch plantearon dos preguntas distintas, variantes del problema original de la Galería de Arte.

Estos fueron llamados: El Problema de la Fortaleza, *The Fortress Problem* ; y El Problema de la Prisión, *The Prison Yard Problem*.

El primero pregunta por el número de guardias necesarios para ver el exterior de un polígono, y el segundo pregunta por el número de guardias necesarios para ver ambos, interior y exterior. El primero de ellos ha sido resuelto de forma satisfactoria, mientras que el segundo aún no se encuentra resuelto.

**Problema de la Fortaleza** El nombre de este problema viene dado por la necesidad de vigilancia de una fortaleza ante posibles ataques. Este problema trata de resolver la pregunta siguiente: ¿Cuántos guardias en vértices se necesitan para ver el exterior de un polígono de  $n$  vértices? Un punto exterior  $y$  es visto por un guardia desde el vértice  $z$  si y sólo si el segmento  $zy$  no intersecta el interior del polígono.

**Teorema 4.1.7 (Teorema de la Fortaleza. O'Rourke y Wood, 1983)**  $\lfloor n/2 \rfloor$  guardias en los vértices son necesarios y suficientes para ver el exterior de un

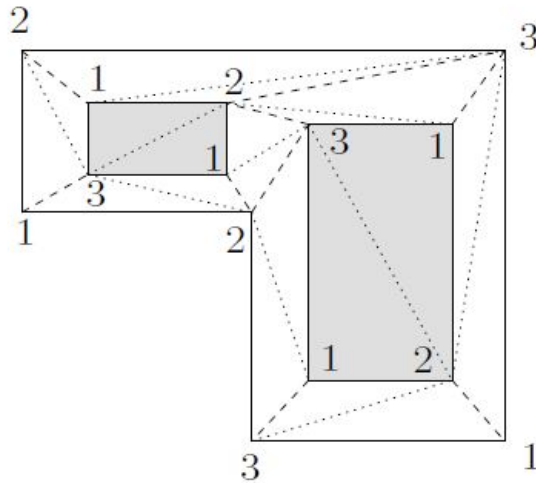


Figura 4.8: Polígono con 2 obstáculos, triangulado y 3-coloreado (Morales, 2006).

polígono de  $n$  vértices, (O'Rourke, 1987).

Dada la planta de un edificio, representada por un polígono  $P$ , se construye el **Cierre Convexo** de dicho polígono, es decir, el mínimo polígono convexo que lo contenga. Tras este paso, se procede a triangular las **Bolsas**, esto es, la porción de plano que está dentro del cierre convexo (casco), pero exterior al polígono. Al grafo resultante de  $n$  nodos se le llamará  $G''$ . Se añade un nodo adicional  $v_\infty$  a  $G''$  fuera del casco y se hace adyacente a cada nodo del casco. A este grafo de  $n+1$  nodos se le llamará  $G'$ . Finalmente, se escogerá algún vértice del casco  $x$  y se separará en dos vértices  $x'$  y  $x''$ , así que el grafo resulta plano. Se añade un nuevo arco tal que  $v_\infty$  sea adyacente a ambos  $x'$  y  $x''$ . Al grafo resultante se le llamará  $G$  (Véase Figura 4.9).

Este grafo resultante es un grafo de triangulación de un polígono. Esto puede ser visto como una apertura del casco convexo en  $x' - x''$ .

Desplazando  $v_\infty$  lo suficientemente lejos, las conexiones podrían interpretarse como líneas rectas. Ya que  $G$  es un grafo de triangulación, puede ser 3-coloreado. El color con menor frecuencia usado, ocurre no más de  $\lfloor (n+2)/3 \rfloor$  veces. No se puede colocar a los guardias en el color que le haya sido asignado a  $v_\infty$ , dado que ese vértice es un punto teórico que existe en el infinito, y que por tanto no es real. Así que los guardias se colocarán en el segundo color menos repetido (Véase Figura 4.10).



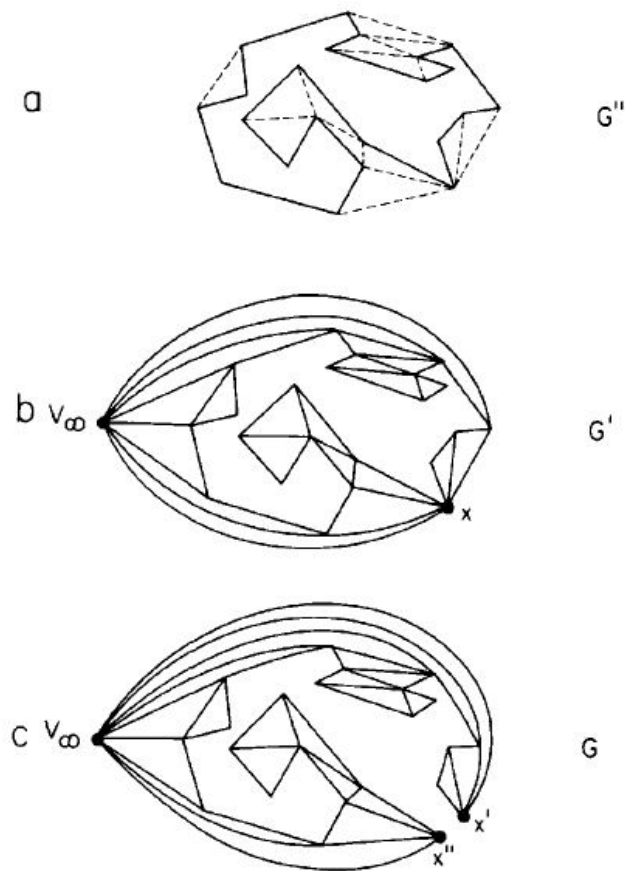


Figura 4.9: El Grafo  $G$  en c) es producido desde la triangulación de las bolsas en a) y conectando todos los vértices del cierre convexo a  $v_\infty$ , en b), se separa algún vértice  $x$  en dos:  $x'$  y  $x''$  (O'Rourke, 1987).

### Algoritmo de la Fortaleza

- Entrada: Polígono  $P$ .
- Salida: Posiciones de los guardias del tal forma que vigilen todo el exterior de  $P$ .
  - Paso 1. Cierre convexo:  
Encontrar el cierre convexo del polígono  $P$ .
  - Paso 2. Triangulación:

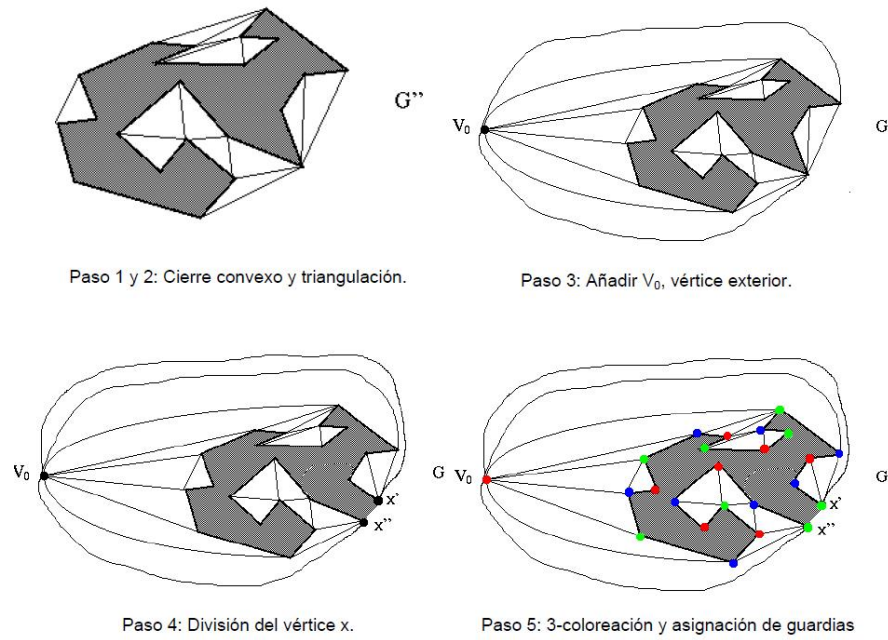


Figura 4.10: Pasos del Algoritmo del Problema de la Fortaleza (O'Rourke, 1987).

Triangular la porción del plano que se encuentra dentro del cierre convexo, pero es exterior a  $P$ . El grafo  $G''$  resultante se compone de  $n$  vértices, correspondientes a los  $n$  vértices de  $P$ .

- Paso 3. Añadir vértice exterior:

Añadir un vértice llamado  $v_\infty$  fuera del cierre convexo de  $P$  y hacerlo adyacente a todos los vértices del contorno. Este nuevo grafo  $G'$  está compuesto por  $[n + 1]$  vértices.

- Paso 4. División de un vértice:

Elegir uno de los vértices del cierre convexo,  $x$ , y separarlo en dos vértices  $x'$  y  $x''$ . Asignar la antigua arista  $x - v_\infty$  a uno de ellos,  $x' - v_\infty$ , y crear una nueva arista desde el otro vértice,  $x'' - v_\infty$ , a fin de que  $x'$  y  $x''$  sean adyacentes a  $v_\infty$ . Este nuevo grafo compuesto por  $[n + 2]$  vértices se llama  $G$ .

Ahora, el interior de  $P$  se convierte en el exterior de  $G$ . Se dice que  $G$  es el grafo de la triangulación de un polígono. Esto puede entenderse como *abrir* el cierre convexo en  $x'$  y  $x''$ .

- Paso 5. 3-coloración y asignación de guardias:

Puesto que  $G$  es un grafo de triangulación de un polígono, puede ser

3-coloreable. El color utilizado con menor frecuencia, por ejemplo el azul, no se produce más de  $\lfloor (n+2)/3 \rfloor$  veces.

Si  $v_\infty$  no es de color azul, entonces la colocación de los guardias en los vértices azules cubre el exterior del polígono original  $P$ .

#### El Problema de la Fortaleza en Polígonos Ortogonales

Aunque son necesarios más guardias para cubrir el exterior que el interior de un polígono arbitrario:  $\lceil n/2 \rceil$  para el exterior *versus*  $\lfloor n/3 \rfloor$  para el interior; con polígonos ortogonales el número requerido difiere sólo levemente:  $\lceil n/4 \rceil + 1$  para el exterior *versus*  $\lfloor n/4 \rfloor$  para el interior.

**Teorema 4.1.8 (Aggarwal, 1983)**  $\lceil n/4 \rceil + 1$  guardias en vértices son necesarios y suficientes para ver el exterior de un polígono ortogonal de  $n$  vértices, (Aggarwal, 1984).

Este Teorema es demostrado con la siguiente prueba: dado un polígono ortogonal de  $n$  vértices,  $P$ , eliminar la arista de mayor coordenada  $y$  (o cualquiera de ellas si hubiera más de una), extender las dos aristas adyacentes verticales y encerrar al polígono completo en un rectángulo, como el que se muestra en la Figura 4.11. El interior de este nuevo polígono  $P'$  de  $n+4$  vértices coincide con el exterior del polígono  $P$ , excepto por el rectángulo  $Q$  sombreado en la figura, el cual es exterior a ambos.

Tras esto, se cuadrilateraliza el polígono resultante, dividiéndolo en cuadriláteros convexos, procediendo de la misma forma que la explicada para Polígonos Ortogonales. A continuación se colorean los vértices de cuatro colores distintos. Aquel color que menos se repita será donde deban ubicarse los guardias.

#### Guardias en el Plano

En esta sección se muestra que  $\lceil n/3 \rceil$  guardias son ocasionalmente necesarios y suficientes para cubrir el exterior de un polígono si los guardias no están restringidos a los vértices del polígono, pudiendo ser localizados en cualquier parte externa al polígono o en el borde del mismo.

Estos guardias reciben la nominación de Guardias en Puntos o en el Plano (*Point Guards*).

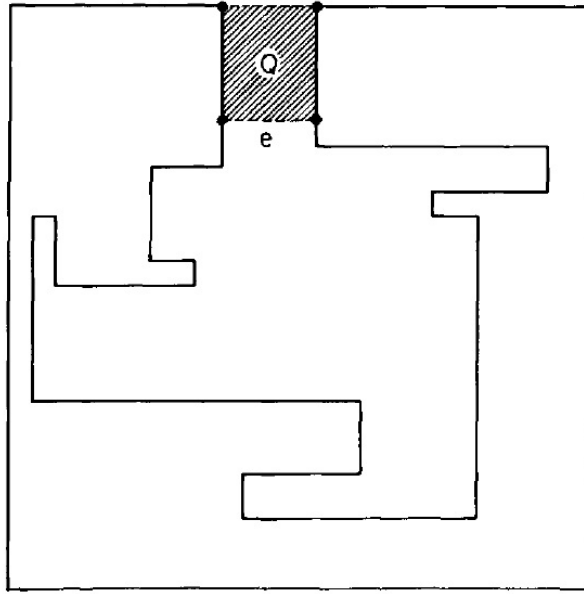


Figura 4.11: El exterior de un polígono ortogonal debe ser convertido en el interior de otro polígono eliminando la arista más alta  $e$  y encerrándolo en un rectángulo (O'Rourke, 1987).

La necesidad de  $\lceil n/3 \rceil$  viene dada por la figura del peine (Véase Figura 4.12).

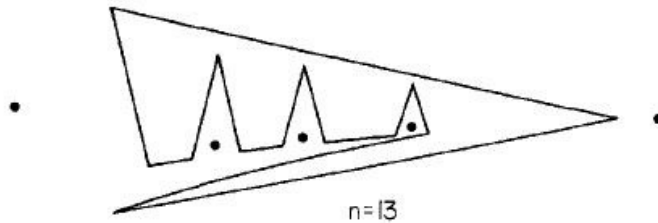


Figura 4.12: Prueba de que  $\lceil n/3 \rceil$  guardias son necesarios para cubrir el exterior de esta figura (O'Rourke, 1987, p.151).

Sea  $P$  un polígono cualquiera (Véase Figura 4.13), rotando  $P$  de forma que el vértice  $a$  es el punto más alto, y  $b$  el más bajo; añadir dos vértices  $\lambda$  y  $\rho$  por debajo del vértice más bajo de  $P$ , y lo suficientemente lejos para que ambos vean a  $a$ . Triangular el interior del cierre convexo exterior al polígono y añadir diagonales desde  $\lambda$  y  $\rho$  a cada vértice perteneciente al casco visible a ellos. Finalmente, separar el vértice  $a$  en dos. El resultado es una triangulación de

un polígono de  $n + 3$  vértices ( $\lambda$ ,  $\rho$  y el vértice creado al separar  $a$  en dos). Al aplicar el Algoritmo de Avis y Toussaint (Apartado 4.1.1.3), es posible cubrir el polígono con  $\lfloor (n + 3)/3 \rfloor = \lceil (n + 1)/3 \rceil$  guardias.

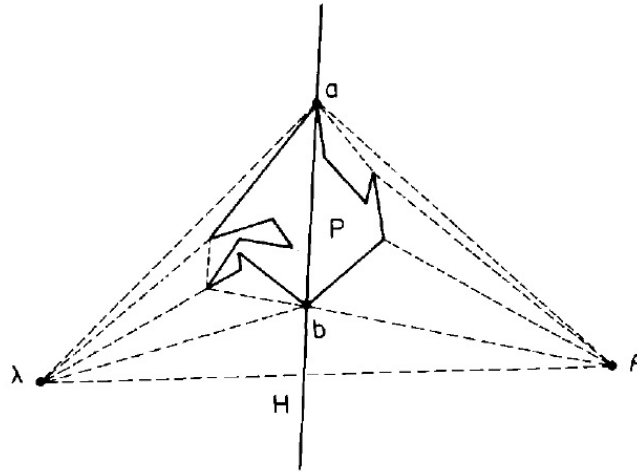


Figura 4.13: Guardias en el exterior del polígono (O'Rourke, 1987, p.151).

**Teorema 4.1.9 (Aggarwal y O'Rourke, 1984)**  $\lceil n/3 \rceil$  guardias en puntos son a veces necesarios y siempre suficientes para cubrir el exterior de un polígono  $P$  de  $n > 3$ , (O'Rourke, 1987).

**Problema de la Prisión** Este problema da respuesta a la pregunta: ¿Cuántos guardias en vértices son necesarios para vigilar simultáneamente el interior y el exterior de un polígono  $P$  de  $n$  vértices?

Un punto interior  $x$  es vigilado por un guardia en un vértice  $z$ , si el segmento  $zx$  no intersecta el exterior de  $P$ , y un punto exterior  $y$  es vigilado por  $z$  si  $zy$  no intersecta el interior de  $P$ . Se consideran a los guardias colocados en vértices.

**Teorema 4.1.10**  $\lfloor \frac{5n}{12} + 2 \rfloor$  guardias en los vértices son suficientes para proteger el interior y exterior de un polígono ortogonal  $P$  con  $n$  vértices, (Hoffmann and Kriegel, 1996).

También se ha desarrollado el problema para polígonos con obstáculos, y para guardias en puntos y no en vértices.

#### 4.1.2.4. Vigilancia con visibilidad de alcance limitado

Hasta el momento, todas estas variantes del Problema de la Galería de Arte han considerado que la visión del vigilante es infinita, y de una amplitud de  $360^\circ$  alrededor del mismo. Esta suposición discrepa de la realidad a la cual se necesita aplicar este tipo de problemas.

Por ello, siguiendo a la tesis de Santiago Canales (Canales Cano, 2004), se hace hincapié en dos conceptos: la visibilidad de alcance limitado, y la  $t$ -buena iluminación.

El primero de estos dos conceptos ya fue introducido por Ntafos en 1992 (Ntafos, 1992), añadiendo la restricción de la distancia máxima a la que un vigilante es capaz de divisar de forma correcta: «dos puntos son  $d$ -visibles si son visibles y su distancia es a lo más  $d$ » (Véase Figura 4.14).

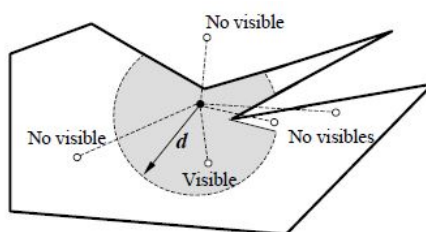


Figura 4.14: Visibilidad de alcance  $d$  (Canales Cano, 2004).

El segundo de los conceptos se introduce por primera vez en la tesis anteriormente citada, y su idea fundamental se basa en que una estructura geométrica sólo está bien iluminada si todos los puntos que la iluminan están bien distribuidos alrededor de ella.

Para dar solución a estas variantes, se desarrollan diferentes métodos heurísticos, utilizando Algoritmos Genéticos, entre otras técnicas de optimización.

#### 4.1.2.5. Miscelaneos

Además de las variantes expuestas en esta sección, se han desarrollado muchas más. Por ejemplo, el problema de vigilar exclusivamente los muros de una galería de arte, desarrollado por Aldo Laurentini en su artículo que tiene por nombre «Guarding the walls of an art gallery» (Laurentini, 1999). También es interesante el problema de optimización de rutas de evacuación (Abellanas

and Hernández, 2007).

Por otro lado, existen variantes del Problema de la Galería de Arte en las cuales los guardias no permanecen fijos en un punto, sino que tienen la propiedad de moverse en línea recta. Esta variante recibe el nombre de **Guardias Móviles** y fue propuesta por Toussaint y O'Rourke (O'Rourke, 1983), demostrando que  $\lfloor \frac{n}{4} \rfloor$  guardias móviles son siempre suficientes y ocasionalmente necesarios para vigilar un polígono  $P$  con  $n \geq 4$  vértices.

Por último, cabe destacar el **Problema del Barrendero**, el cual consiste en buscar el camino que debe seguir un barrendero, para que sea capaz de cubrir de forma completa la superficie de un polígono  $P$ , utilizando la escoba a su alrededor, con un alcance limitado  $d$ .

##### 4.1.3. Vigilancia en tres dimensiones

Volviendo al tema que se aborda en la presente investigación, resulta lógico pensar que efectivamente, esta variante del problema es lo que más interesa estudiar: vigilar objetos tridimensionales de forma óptima.

Sin embargo, como bien menciona Óscar Morales Ponce <sup>1</sup>, en su tesis *Vigilando Poliedros Ortogonales 3D* (Morales, 2006), dirigida por el profesor Jorge Urrutia: «...diversas variaciones del Problema de la Galería de Arte han aparecido en la literatura, entre éstas se encuentran, vigilar polígonos ortogonales, polígonos con obstáculos, usar guardias móviles, etcétera. Sin embargo, pocos resultados han sido obtenidos en dimensiones mayores, entre los más notables se encuentra vigilar terrenos poliédricos».

No por ello se debe dejar de mencionar los resultados obtenidos en esta variante, los cuales se enumeran a continuación:

- Prueba de que la superficie de cualquier poliedro ortogonal 3D con  $n$  vértices puede ser vigilado con a lo más  $\lfloor \frac{n}{3} \rfloor$  guardias en los vértices del poliedro. Esta cota no es justa, es decir, se puede mejorar.
- Prueba de que la superficie de cualquier poliedro ortogonal 3D con  $m$  aristas puede ser vigilado con a lo más  $\lfloor \frac{m}{6} \rfloor$  guardias colocados en las aristas del poliedro. Para este problema se obtuvo la cota justa.

---

<sup>1</sup>Universidad Nacional Autónoma de México

- Prueba de que el interior de cualquier poliedro ortogonal 3D con  $m$  aristas puede ser vigilado con a lo más  $\lfloor \frac{m}{6} \rfloor$  guardias colocados en las aristas del poliedro. Esta cota no es justa, por lo que el problema queda abierto.
- Prueba de que el exterior de cualquier poliedro ortogonal 3D con  $m$  aristas puede ser vigilado con a lo más  $\lfloor \frac{m}{6} \rfloor$  guardias colocados en las aristas del poliedro. Esta cota no es justa, por lo que el problema queda abierto.

#### 4.1.4. Conclusión

Una vez analizados los diferentes Problemas de Iluminación, es posible observar que ninguno de los casos expuestos describe exactamente el problema propuesto en la presente tesis: la vigilancia del borde exterior de un edificio desde el lado externo al mismo.

Sin embargo, de entre todos los problemas estudiados, el que más se ajusta a la realidad con la que se encuentran los profesionales de la Topografía a la hora de decidir desde qué puntos realiza la toma de datos para ejecutar un levantamiento arquitectónico, es el Problema de la Fortaleza. Éste es el único que relaciona el exterior y el borde de un polígono. También resulta muy interesante tener en cuenta la *Visibilidad de Alcance Limitado* (Ntafos, 1992), así como la *Vigilancia de Polígonos con Obstáculos* (Apartado 4.1.2.1).

Más adelante, se tratará de adaptar estas técnicas, diseñando heurísticas que den solución al problema propuesto, el cual es de naturaleza *NP-duro*.

#### 4.1.5. Aplicación del Problema de la Galería de Arte a los Levantamientos Arquitectónicos

La investigación llevada a cabo por los profesores Héctor González Baños y Jean Claude Latombe denominada «A Randomized Art-Gallery Algorithm for Sensor Placement» (González-Baños, 2001), ha resultado de gran interés para la presente investigación, pues en ella se aplica el Problema de la Galería de Arte a tratar de optimizar los puntos desde los cuales realizar la toma de datos con instrumentos como el escáner láser.

En el artículo citado anteriormente se describe una estrategia de ubicación para computar un conjunto de buenos puntos de vista interiores, desde los cuales la toma de datos sea más efectiva. Además, el edificio se representa como un polígono definido por la planta del mismo, resolviendo una variante



del Problema de la Galería de Arte, en la cual se añaden ciertas restricciones. En este caso los guardias no tienen una visión infinita, sino que tienen un rango mínimo y máximo de trabajo, y un ángulo mínimo de incidencia a partir del cual obtiene buenos resultados.

Se insiste en que la adquisición de datos tridimensionales de alta calidad es una operación costosa, y su post-proceso puede ser computacionalmente caro. Por lo que se piensa que es totalmente necesario minimizar el número de posicionamientos desde los cuales realizar la toma de datos, de forma que la nube de puntos sea lo menos pesada posible. El problema de vigilar el interior de un polígono está estrechamente relacionado con el Problema de la Galería de Arte.

Para continuar, en esta investigación se especifica que lo que se busca vigilar es solamente el contorno del edificio, y el contorno de los obstáculos que puedan existir en su interior, producidos por objetos tales como pilares.

Como puede verse en la Figura 4.15 el algoritmo consiste en dividir en celdas las aristas a vigilar. Cada uno de los puntos de vista candidatos son relacionados con las celdas que son capaces de vigilar

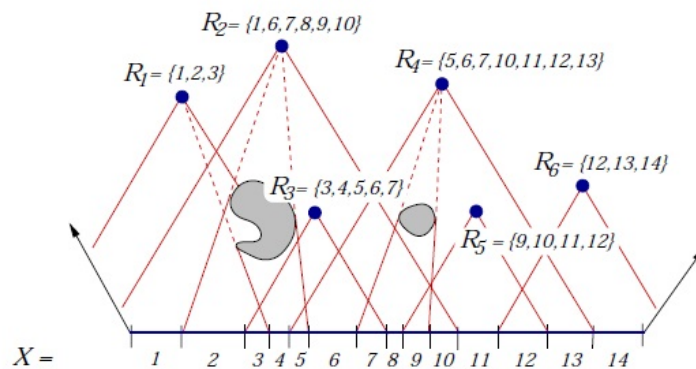


Figura 4.15: Cada arista es dividida en celdas, y a cada punto de vista se le asignan las celdas que vigila correctamente (González-Baños, 2001).

En este artículo, la optimización se realiza a través de un *Algoritmo Voraz* cuya definición aparece en el Apartado 1.5.3.1.

Como puede verse en la Figura 4.16, el número de guardias es menor cuando no existe una limitación en el ángulo de incidencia a partir del cual se vigila correctamente, frente a cuando el ángulo de incidencia es limitado.

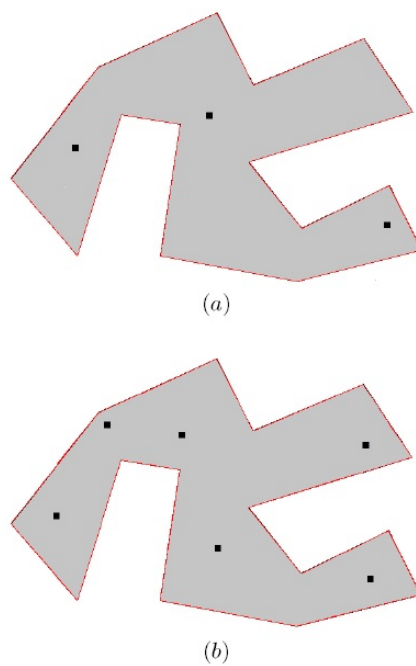


Figura 4.16: a) Tres guardias son suficientes para vigilar el contorno interior del polígono sin limitación del ángulo de incidencia. b) Seis guardias son necesarios cuando el ángulo de incidencia está limitado (González-Baños, 2001).

### 4.2. De los Algoritmos Genéticos (AG)

Anteriormente, en el Apartado 1.5.3.2 se ha situado a los Algoritmos Genéticos dentro de la IA. En este apartado se desarrolla de una forma amplia el funcionamiento de esta técnica de optimización metaheurística.

#### 4.2.1. Introducción

La Teoría de la Selección Natural expone que las plantas y los animales que existen hoy en día son el resultado de millones de años de adaptación a las demandas medioambientales (Sivanandam and Deepa, 2007). Las Técnicas de Computación Evolutivas convierten estos principios evolutivos en algoritmos que pueden ser usados para la búsqueda de soluciones óptimas a un problema.

Cuando el espacio de búsqueda de una solución óptima es pequeño, se pueden evaluar todas las posibles soluciones en un tiempo razonable. Sin embargo, cuando el espacio de búsqueda se extiende, esta técnica se convierte en impracticable. En la búsqueda evolutiva se encuentra una técnica mejor que la búsqueda aleatoria.

Como se puede leer en el libro *Introducción a los Algoritmos Genéticos y la Programación Genética* (Gestal et al., 2010) el desarrollo de esta técnica se debe, en gran medida, a John Holland, investigador de la Universidad de Michigan. Dicho investigador, a finales de la década de los sesenta, desarrolló esta técnica, que originalmente fue denominada *Planes Reproductivos*, y tras la publicación de su libro *Adaptation in Natural and Artificial System* (Holland, 1992) se le atribuye el nombre de **Algoritmos Genéticos**.

Traduciendo las palabras de David E. Golberg, estos algoritmos combinan la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado, aunque aleatorizado, para constituir así un algoritmo de búsqueda que tenga algo de las genialidades humanas (Golberg, 1989).

Aunque los detalles de la evolución biológica no son completamente comprensibles, incluso hoy en día, existen ciertos puntos bastante evidentes (Sivanandam and Deepa, 2007):

- La Evolución es un proceso que opera sobre cromosomas, en lugar de operar sobre los organismos.

- La Selección Natural es un mecanismo que relaciona el cromosoma con la eficiencia de la entidad a la que representa, permitiendo a los organismos eficientes que están bien adaptados al medio, a reproducirse con más frecuencia que aquellos que no lo están.
- El proceso evolutivo tiene lugar durante el periodo de reproducción. Existe un gran número de mecanismos de reproducción en la naturaleza. Algunos de los más comunes son la mutación y la recombinación.

De esto se puede obtener la conclusión de que aquellos individuos que peor se adaptan al medio, tienen menos posibilidades de crear descendencia. De esta manera, se entiende que los genes de los individuos mejor adaptados se propagarán con más probabilidad en las siguientes generaciones. De lo cual se deduce que las generaciones futuras heredarán las buenas cualidades de sus predecesores, creándose generaciones cada vez más adaptadas al medio.

#### 4.2.2. Funcionamiento de un Algoritmo Genético

Un Algoritmo Genético, AG, parte de un conjunto inicial de posibles soluciones al problema. Este conjunto de soluciones son los denominados **Individuos** y entre ellos conforman la **Población**. La primera población recibe por nombre **Población Inicial**. El **medio** en el que viven estos individuos es el problema que se intenta resolver. Por lo que los individuos se irán adaptando al medio generación tras generación.

Cada posible solución, o individuo, lleva asignado un **código**, el cual se puede asemejar al cromosoma del individuo, o dicho de otra forma, el ADN del individuo. Este código es el que hace a un individuo distinto al resto y a partir de dicho código se pueden conocer todas sus características.

En un Algoritmo Genético, los individuos representan posibles soluciones al problema. Éstos pueden representarse como un conjunto de parámetros, que se denominan genes. Estos genes se agrupan formando una  $n$ -upla de valores, a menudo referida como cromosoma. A una colección de  $n$  números reales dados en un cierto orden se llama una  **$n$ -upla**.

Estos cromosomas tendrán asociado un **valor** que cuantifica la validez de un individuo como solución al problema, es decir, cómo de idóneo es el individuo para el medio en el que vive. El punto más importante en el momento de diseñar un algoritmo genético es el de diseñar una función matemática capaz

## 4.2. DE LOS ALGORITMOS GENÉTICOS (AG)

---

de cuantificar las cualidades de estos individuos a través de un valor. Esta función tiene el nombre de **Función Objetivo** (también puede encontrarse como Función *Fitness* o Función Adaptación).

Las labores más importantes a la hora de diseñar un Algoritmo Genético son dos: la codificación de los individuos, y el diseño de la Función Objetivo:

- Codificación: de ella debe poder obtenerse la información necesaria para que la evaluación del individuo sea efectiva.
- Diseño de la Función Objetivo: debe servir para obtener valores fiables que den la posibilidad de ordenar a todos los individuos según la validez que tengan como solución al problema.

Cada nueva generación sustituirá a la anterior e incorporará individuos que hereden las buenas cualidades de sus padres, haciendo que las generaciones sean cada vez más competitivas y adaptadas al medio.

El algoritmo genético debe haber sido correctamente diseñado para que se detenga cuando la población converja hacia una solución óptima.

Como se refleja en el diagrama de flujo de la Figura 4.17, un AG comienza con la generación de la Población Inicial. Esa población es evaluada. En caso de que se alcanzase algún criterio de parada, se obtendría la Población Óptima, y el algoritmo finalizaría. En caso de que no, se procede a la Selección de los mejores individuos, tras ello se crearán los nuevos individuos con las operaciones de Cruce y de Mutación. La nueva generación es evaluada, y de nuevo vuelve a analizarse si se alcanza o no algún criterio de parada.

### 4.2.3. Diferencias entre los Algoritmos Genéticos y los métodos convencionales de optimización

En una publicación de los profesores Majeed y Kumar, donde se hace una revisión de los AG aplicados a la ingeniería, se ponen de manifiesto las ventajas que éstos presentan (Majeed and Kumar, 2014).

- Los métodos convencionales de optimización operan en los parámetros del problema directamente, mientras que los algoritmos genéticos operan sobre una versión codificada de los parámetros del problema.

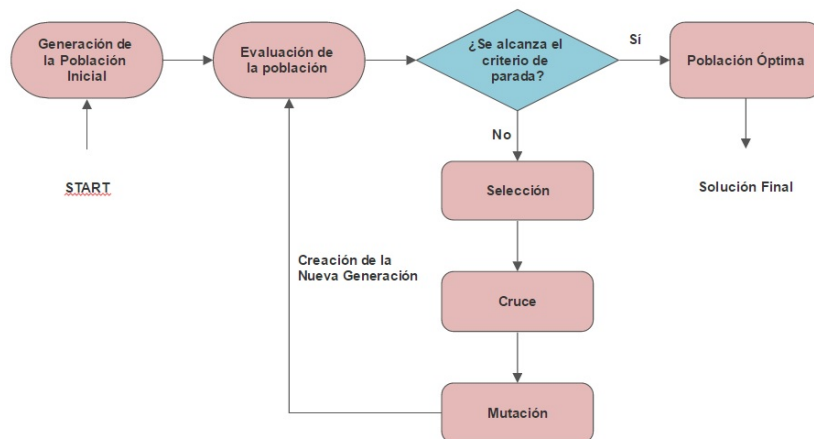


Figura 4.17: Gráfico de Flujo de un Algoritmo Genético. Elaboración Propia, Fuente (Majeed and Kumar, 2014).

- La mayoría de los métodos convencionales operan sobre una única solución para producir una solución óptima, mientras que los algoritmos genéticos operan sobre una población de soluciones, seleccionando soluciones cada vez más óptima en cada generación.

A continuación se exponen las ventajas que presentan los AGs, frente a otras técnicas de optimización (Majeed and Kumar, 2014):

1. Poseen una gran capacidad para el procesamiento en paralelo.
2. Proporcionan un espacio de solución más amplio.
3. Ofrecen posibilidad de encontrar fácilmente mínimos globales.
4. Son fáciles de modificar.
5. Son capaces de manejar funciones con ruido de forma eficiente.
6. Muestran un gran rendimiento, incluso en problemas multi-modales.
7. No necesitan conocimientos previos del espacio del problema.
8. No se ven muy afectados por las discontinuidades del espacio del problema.

9. Son lo suficientemente confiables como para no quedar atrapados en mínimos locales.

Para continuar, se procede a enumerar las limitaciones de los algoritmos genéticos:

1. No es fácil encontrar una función objetivo.
2. En muchos casos, convergen prematuramente a una buena solución.
3. Es una tarea complicada elegir los parámetros óptimos para su diseño.
4. Necesitan de una técnica de búsqueda local que funcione de forma efectiva.
5. Necesitan un gran número de evaluaciones de la función objetivo.
6. No es fácil configurar un sistema basado en un algoritmo genético.

### 4.2.4. Creación de la Población

#### 4.2.4.1. Población inicial

El primer paso que se ejecuta en un AG es la creación de la **Población Inicial**. Usualmente, los individuos que conforman la población inicial son creados de forma aleatoria. Existen casos en los que la población inicial es creada a partir de alguna técnica anterior de optimización (Moujahid et al., 2008). Esto aceleraría la convergencia del algoritmo, y es una técnica usada con frecuencia. Aunque puede presentar la desventaja de provocar una convergencia demasiado temprana del algoritmo.

Para diseñar un AG hay que tener en cuenta que esta técnica de optimización funciona por sí sola, si está correctamente diseñada. Por tanto, no es recomendable prefijar, o limitar demasiado el espacio de búsqueda. El algoritmo genético es espontáneo, por lo que no es conveniente *darle demasiadas pistas*. Si los elementos que lo conforman están pensados de la forma correcta, las generaciones irán adaptándose poco a poco al medio en el que viven, como ocurre con las especies de la naturaleza.

El tamaño de la Población Inicial debe ser tal que sea lo suficientemente grande como para asegurar la diversidad, pero no tan grande como para que el algoritmo no sea eficiente (Bajuelos Domínguez et al., 2008).

De lo que se deduce que el tamaño de la población debe ser proporcional a la naturaleza del problema. Alander (Alander, 1992), sugiere que una población de entre  $l$  y  $2l$  puede funcionar, siendo  $l$  el tamaño de la ristra de valores que da código a un individuo.

#### 4.2.4.2. Codificación de los individuos de la población

A continuación se identifica la estructura de los individuos de una población en un AG (Véase Figura 4.18):

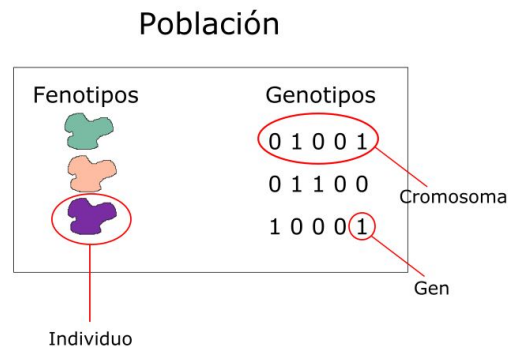


Figura 4.18: Esquema de los elementos que componen un Algoritmo Genético. Elaboración Propia.

- **Población:** los distintos individuos conforman una población. Las poblaciones van cambiando generación tras generación.
- **Individuo o fenotipo:** un individuo representa una posible solución al problema que se desea optimizar. También se le puede denominar como fenotipo.
- **Gen:** cada uno de los parámetros que compone un cromosoma es un gen.
- **Cromosoma:** el conjunto de todos los parámetros que caracteriza a un individuo se codifica en una cadena de valores, la cual recibe el nombre de cromosoma.
- **Genotipo:** el conjunto de los parámetros representados por un cromosoma particular, se denomina genotipo.



Si bien, lo usual es encontrar a los individuos codificados en código binario, no todos los problemas pueden medir sus variables de esta forma. Puede ocurrir que un individuo se codifique según distintas variables cuyo valor no sea cero o uno. Esto dependerá por tanto de la naturaleza del problema a estudiar.

Dado un problema, el cual se desea optimizar mediante un AG, se deben codificar los individuos de forma que, a través del código, se pueda fácilmente identificar al individuo. Además, del genotipo de un individuo se debe poder sacar toda la información necesaria para evaluar la calidad de dicho individuo.

Por lo que la codificación debe permitir dos cosas: diferenciar de forma inequívoca a un individuo con respecto a otros y extraer la información necesaria para conocer la idoneidad con la que el individuo se adapta al medio en el que vive.

### 4.2.5. Evaluación de la Población

Para evaluar la población es necesario definir dos conceptos: la Función Objetivo y el Valor de Ajuste. La Función Objetivo evalúa la idoneidad con la que un individuo se adecua al medio en el que vive, devolviendo un valor. Ese valor es lo que se conoce como **Valor de Ajuste** (también puede ser llamado *Fitness*). En función a él, el individuo tendrá más o menos oportunidades de supervivencia y/o de crear descendencia que hereden sus buenas cualidades.

#### 4.2.5.1. Función Objetivo

Para el correcto funcionamiento de un AG se debe diseñar una forma de medir si los individuos de la población representan o no buenas soluciones al problema planteado.

La **Función Objetivo** (también puede encontrarse como *Función Evaluación* o *Función Fitness*) es la encargada de evaluar cada uno de los parámetros que hacen a un individuo más o menos apto.

A cada uno de los parámetros que se tienen en cuenta, se le asigna un valor óptimo. Se trata, por tanto, de crear generaciones cuyos individuos se acerquen cada vez más a estos valores. Es muy probable que dicho valor óptimo no sea alcanzable, pero, tratándose de una técnica metaheurística, bastará con conseguir un buen resultado.

Existen numerosos criterios para diseñar la función objetivo. Dependiendo de la naturaleza del problema, se deberán tomar diferentes decisiones para diseñar correctamente esta función. Como ejemplo, hay casos en los que la función objetivo devalúa el valor de un individuo si éste es muy similar a otro individuo, puesto que no favorece a la diversidad. También puede darse el caso de que un individuo no presente una solución real al problema, pero se maneja como un individuo más ya que puede crear descendencia que sí sea una solución al problema.

#### 4.2.5.2. Valor de Ajuste

El **Valor de Ajuste** (o *Fitness*) de un individuo es el valor que se le asigna tras haber aplicado sobre él la Función Objetivo. Al tratarse de un valor numérico, resulta cómodo ordenar a la población según este valor. Dependiendo de cómo se haya diseñado el AG, este valor puede tender a un mínimo, o a un máximo.

Aunque lo que se busca en un AG es el individuo que mejor valor de ajuste tenga, también es un factor a tener en cuenta el valor de ajuste medio de la población. La razón de ello es que se trata de un indicador que revela la evolución de la especie que se ha creado. Esto quiere decir que un AG en el cual, tras varias generaciones, el valor de ajuste medio de la población no ha mejorado, se puede considerar que el AG se ha estancado.

#### 4.2.6. Operadores Genéticos

Una vez que se ha evaluado la población, será necesario crear una nueva generación. Este proceso es posible gracias a los **Operadores Genéticos**, que son los siguientes: Selección, Cruce, Copia y Mutación.

##### 4.2.6.1. Selección

Mediante la **Selección** se escoge a los mejores individuos de la población. Estos individuos seleccionados, al ser los que mejor se adaptan al medio, tienen muchas posibilidades de, o bien generar descendencia, o bien ser *longevos* y continuar existiendo en la siguiente generación, o ambas opciones.

Como ya se ha comentado anteriormente, los individuos que más proba-

bilidad de reproducirse tienen serán aquellos que estén mejor adaptados al medio. Por tanto, la selección de un individuo está relacionada con su valor de ajuste.

La función de selección más común es la denominada *función de selección proporcional a la función objetivo*.

Sin embargo, si no se le da oportunidad a los individuos menos aptos, la población se convertiría en homogénea muy rápidamente, convergiendo así en pocas generaciones.

Se distinguen varios algoritmos de selección: por Torneo, por Truncamiento, Selección por Ranking Lineal, por Ranking Exponencial y Proporcional o por Ruleta (Blickle and Thiele, 1996). Los más comunes son la Selección Proporcional, o Selección por Ruleta, y la Selección por Torneo.

### Selección por Ruleta

Propuesto por DeJong, es posiblemente el método más utilizado desde los orígenes de los Algoritmos Genéticos (Blickle and Thiele, 1996).

A cada individuo de la población se le asigna una porción de ruleta, en proporción a su valor de ajuste. Haciendo girar la ruleta, es más probable que pare en el individuo que más espacio esté ocupando (Véase Figura 4.19).

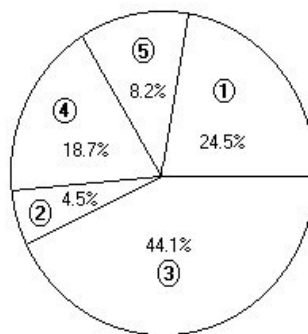


Figura 4.19: Selección por Ruleta (Ramírez, 2008).

### Selección por Torneo

Este método de selección consiste en escoger a un individuo de entre varios, y comparar sus genotipos. Normalmente se enfrentan dos individuos y se escoge al que posea un mejor valor de ajuste.

Los individuos que resulten seleccionados pasan a ser parte de un subgrupo de la población, sobre el cual se pueden realizar las operaciones de cruce y copia.

#### 4.2.6.2. Cruce

Una vez seleccionados los individuos se procede a generar la siguiente generación. El cruce simula a la reproducción sexual.

El proceso de cruce es el siguiente:

1. De entre los individuos seleccionados, se escoge aleatoriamente a dos.
2. A la pregunta de si procede el cruce entre los individuos escogidos en el paso anterior, se le aplica la **Probabilidad de Cruce**,  $P_c$ , la cual suele ser elevada, mayor al 80 %.
3. Tras aplicarse la  $P_c$  pueden darse dos casos:
  - a) Sí procede el cruce: se crean dos nuevos individuos, a partir del método de cruce elegido.
  - b) No procede el cruce: no se crean nuevos individuos.
4. Ambos individuos escogidos en el Punto 1, vuelven al subgrupo de individuos seleccionados.
5. Se recuenta el número de nuevos individuos que se han creado para la siguiente generación.
  - a) Si se ha alcanzado el número de individuos deseados para la próxima generación, se termina la operación de cruce.
  - b) En caso contrario, se itera el proceso partiendo del Punto 1.

Este proceso se repite hasta haber generado el número de nuevos individuos que se desee.

Se entiende que el fruto de la reproducción de dos individuos adaptados al medio, será un individuo que herede las mejores características de los padres.

Existen diferentes algoritmos de cruce. Los más utilizados son el cruce en un punto, el cruce en dos puntos y el cruce uniforme.

- Cruce en un punto (Véase Figura 4.20): También denominado SPX (*Single Point Exchange*). Una vez seleccionados los padres, se selecciona aleatoriamente un punto, por el cual se corta a sus cromosomas. Se intercambian las colas entre los dos individuos para generar los nuevos descendientes.

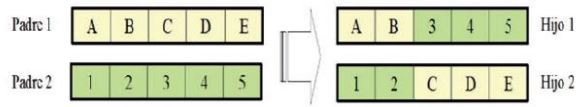


Figura 4.20: Cruce en un punto (Gestal et al., 2010, p.21).

- Cruce en dos puntos (Véase Figura 4.21): También denominado DPX (*Double Point Crossover*). Funciona como el cruce en un punto, pero en este caso se corta por dos puntos. Ninguno de los puntos podrá ser un extremo. Esta técnica puede tener la desventaja de que al dividir en tantos tramos el cromosoma del individuo, puede perderse la bondad que poseían los genes anteriormente.

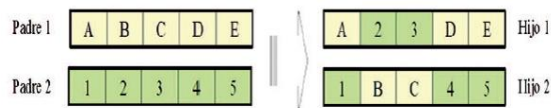


Figura 4.21: Cruce en dos puntos (Gestal et al., 2010, p.22).

- Cruce uniforme (Véase Figura 4.22): El cruce uniforme es una técnica en la que cada gen de la descendencia tiene las mismas probabilidades de pertenecer a un padre o a otro. La descendencia contiene una mezcla de los genes de los padres. La máscara se genera de forma aleatoria. La probabilidad de que se produzca una ruptura de los buenos esquemas crece al aumentar el número de puntos de cruce.

### 4.2.6.3. Mutación

La **Mutación** de un individuo es la variación de alguno de sus genes de forma aleatoria. La mutación se considera un operador básico, que proporciona un pequeño elemento de aleatoriedad en la vecindad de los individuos de la población. El operador de mutación va ganando en importancia a medida que la población de individuos va convergiendo (Moujahid et al., 2008).

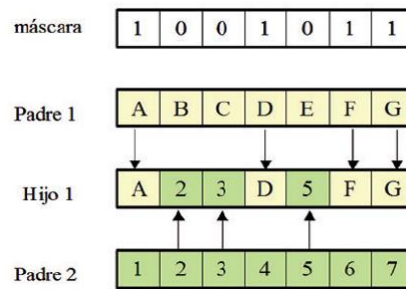


Figura 4.22: Cruce Uniforme (Gestal et al., 2010, p.23).

La mutación suele realizarse de forma conjunta con la operación de cruce: una vez generados los nuevos individuos tras el proceso de cruce descrito anteriormente en el Paso 3a, se procede de la siguiente forma:

1. Se generan dos nuevos individuos:  $I_1$  e  $I_2$ .
2. Se aplica sobre  $I_1$  la **Probabilidad de Mutación**,  $P_m$ , la cual suele ser bastante baja, alrededor del 1 %, pero puede ser mayor.
3. Tras aplicarse la  $P_m$  pueden darse dos casos:
  - a) Sí procede la mutación: se modifica de forma aleatoria alguno de los genes que componen su genotipo.
  - b) No procede la mutación: el individuo permanece intacto.
4. Se repite el proceso con el  $I_2$ .
5. Fin del proceso de mutación.

Se ha demostrado que un algoritmo genético en el que se introduzca el operador de la mutación evoluciona mejor que un algoritmo basado solamente en la selección y el cruce (Schaffer et al., 1989).

En la mayoría de los algoritmos genéticos, las tasas de reproducción y de mutación se mantienen constantes durante generaciones. No obstante, algunos autores han mejorado los resultados modificando la probabilidad de mutación a medida que aumenta el número de iteraciones (Ackley, 2012), (Bramlette, 1991), (Fogarty, 1989), (Michalewicz and Janikow, 1991).

### 4.2.6.4. Copia

Consiste simplemente en la copia de un individuo de una generación a la siguiente. El porcentaje de copia que se da en una población suele ser bajo. En la realidad simula al comportamiento de aquellos seres vivos *longevos*.

El **Elitismo** es un caso particular del operador de copia consistente en copiar siempre al mejor, o mejores, individuos de una generación a la siguiente. De esta forma nunca se dará un paso atrás en la calidad de la población de una nueva generación.

### 4.2.6.5. Creación de la siguiente generación

Una vez que se hayan creado los nuevos individuos siguiendo el proceso de Cruce y Mutación, y tras haber realizado, si así se hubiera decidido, la operación de Copia de los mejores individuos de la generación actual, se debe establecer quién va a formar parte de la siguiente generación.

En el caso de que al generar una nueva generación, ésta segunda ocupe el lugar de la primera, se debe establecer el criterio con el que los nuevos individuos van a situarse en la nueva generación. Puede ser de forma aleatoria, reemplazando a los padres, reemplazando a los que tengan un valor de ajuste similar, o reemplazando a los peores.

### 4.2.7. Convergencia

Si el algoritmo ha sido correctamente diseñado, la población ha de evolucionar generación tras generación, alcanzando poco a poco valores de ajuste que se aproximen cada vez más al óptimo. El valor de ajuste medio de la población también debe ir mejorando (Véase Figura 4.23).

El algoritmo finalizará cuando se dé alguna de las siguientes circunstancias:

- Se ha hallado un individuo óptimo, o muy cercano al óptimo. Los mejores individuos de la población representan soluciones suficientemente buenas, acercándose bastante al valor óptimo.
- La población ha convergido. Esto ocurre cuando el 95 % de la población tiene el mismo valor en su codificación.

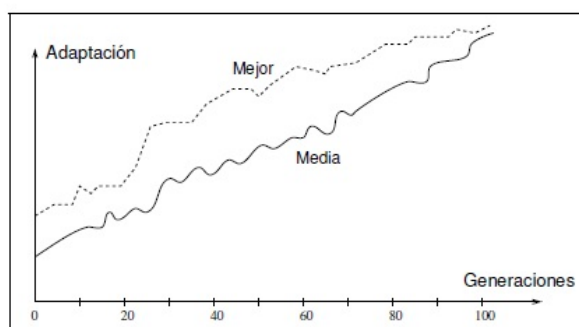


Figura 4.23: Adaptación media y mejor adaptación en un Algoritmo Genético (Moujahid et al., 2008).

- Se ha alcanzado el número de generaciones (iteraciones) máximo que se establece al comienzo del algoritmo. Este valor es considerablemente alto.

Si tras crear una nueva generación se da alguno de los casos anteriores, el algoritmo llegará a su fin. De lo contrario, volverá a crearse una nueva generación a partir de los procesos de selección, cruce, mutación y copia.

#### 4.2.8. Algoritmos Genéticos aplicados al Problema de la Galería de Arte

En el artículo denominado «Optimizing the Minimum Vertex Guard Set on Simple Polygons via a Genetic Algorithm» (Bajuelos Domínguez et al., 2008), se desarrolla un AG con el fin de encontrar el mínimo número de guardias en los vértices para cubrir la vigilancia interior de polígonos simples.

Como ya se ha expuesto en la Sección 1.4, el problema de encontrar el número mínimo de guardias en vértices (*MVG-Problem*) es *NP-duro*.

Mediante el AG diseñado se concluye que el número de guardias necesarios para polígonos irregulares, y para polígonos ortogonales es de  $n/6,38$  y  $n/6,40$  respectivamente, frente a los  $\lfloor n/3 \rfloor$  y  $\lfloor n/4 \rfloor$  que fueron demostrados por los Teoremas 4.1.1 y 4.1.4, siendo  $n$  el número de vértices.

La codificación de los individuos se realiza mediante una cadena de valores binarios, en los que cada valor representa a un vértice. En el caso de que se ubique a un guardia en un vértice, se refleja con un 1, y en caso de no existir



ningún guardia en el vértice, queda reflejado con un 0 (Ver Figura 4.24).

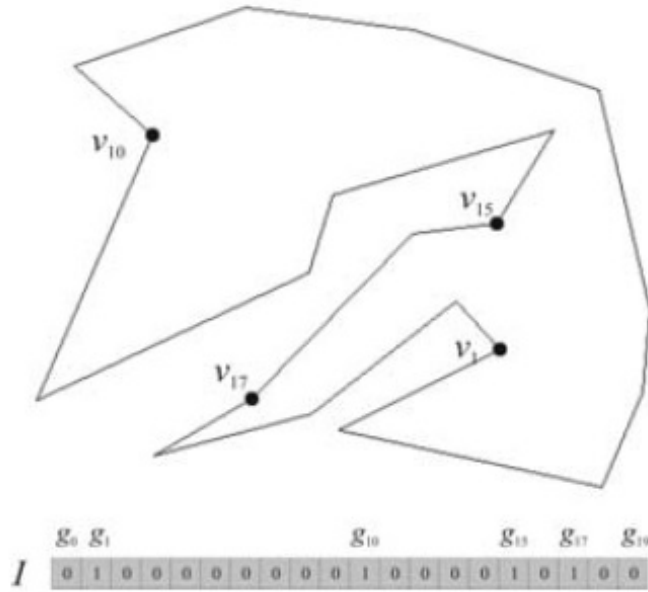


Figura 4.24: Un individuo  $I$  de un polígono arbitrario de  $n = 20$  vértices (Bajuelos Domínguez et al., 2008).

En este algoritmo se tiene en cuenta el número de vértices que son *reflex*, esto es, que el ángulo interior de los mismos es mayor a  $\pi$ .

Un individuo es válido si es capaz de vigilar el interior del polígono de forma completa, por lo que la Función Objetivo tan solo trata de minimizar el número de guardias  $g_j$ .

$$f(I) = \sum_{j=0}^{n-1} g_j \quad (4.1)$$



## Capítulo 5

# Diseño de Métodos de Optimización en el Posicionamiento para la Realización de un Levantamiento Arquitectónico

En este capítulo se desarrolla el proceso de diseño de diferentes métodos de optimización en el posicionamiento para la realización de un Levantamiento Arquitectónico, a través de la aplicación de técnicas deterministas por un lado, y metaheurísticas, por otro.

En primer lugar, se desarrolla el método diseñado basado en el Problema de la Galería de Arte, y en concreto, haciendo uso del Problema de la Fortaleza (Sección 5.1). El método que se diseña es un algoritmo determinista. Esto quiere decir que fijados los datos de entrada, se genera una solución unívoca.

En segundo lugar, se desarrolla un algoritmo basado en técnicas evolutivas, tratándose en concreto de un algoritmo genético. El espacio de búsqueda aumenta teniéndose en cuenta un mayor número de restricciones. En este caso, se trata de un algoritmo no determinista, pues para unos datos de entrada concretos, pueden obtenerse distintas soluciones de salida. El desarrollo de este segundo, a su vez, se expone en dos pasos: el estudio teórico previo (Sección 5.2), y una segunda versión del mismo implementada (Sección 5.3).

## 5.1. Método Determinista.

Tras haber realizado un recorrido por los distintos **Problemas de Visibilidad** en la Sección 4.1, se ha llegado a la conclusión de que la variante que más se asemeja al problema planteado en el presente trabajo de investigación es el denominado **Problema de la Fortaleza**, detallado en el Apartado 4.1.2.3.

### 5.1.1. Similitudes con el Problema de la Fortaleza

El Problema de la Fortaleza y el problema objeto de estudio confluyen en que en ambos se relaciona el borde del polígono con el exterior del mismo. En el primer problema, se desea vigilar el exterior de un polígono, desde el borde del mismo, incluyendo también, por tanto, la vigilancia del propio borde del polígono. En cambio, en el problema planteado se desea vigilar el borde de un polígono desde el lado exterior al mismo.

Recordando el Teorema 4.1.7,  $\lfloor n/2 \rfloor$  guardias en los vértices son necesarios y suficientes para ver el exterior de un polígono de  $n$  vértices, (O'Rourke, 1987). Recuérdese que el Algoritmo de la Fortaleza 4.1.2.3, ubica a los guardias en los vértices del polígono, garantizando que el exterior es completamente vigilado.

En la Figura 5.1 puede apreciarse un polígono no convexo de 19 vértices, al que se le ha aplicado el Algoritmo de la Fortaleza 4.1.2.3.

No obstante, se debe señalar la gran diferencia entre el Problema de la Fortaleza y el problema planteado en el presente trabajo de investigación. Se trata de que en el primero, los guardias se sitúan justo sobre los vértices del edificio; mientras que en el segundo se desean obtener puntos de vigilancia situados en el exterior del polígono.

### 5.1.2. Descripción del Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Caso General

En el algoritmo diseñado y desarrollado en esta sección se localizan las áreas que garantizan una completa vigilancia del borde de un polígono desde el exterior del mismo.

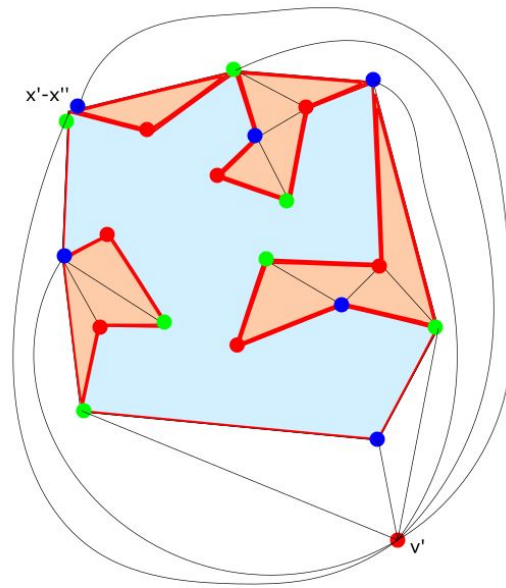


Figura 5.1: Aplicación del algoritmo del Problema de la Fortaleza sobre un polígono. Elaboración Propia.

Este algoritmo parte del Algoritmo de la Fortaleza desarrollado en el Apartado 4.1.2.3, añadiendo el paso por el cual se obtienen las áreas de visibilidad a lo largo de las cuales los guardias obtenidos pueden desplazarse sin perder la visibilidad de aquella porción del borde del polígono que le ha sido asignada.

ENTRADA: Polígono de  $n$  vértices.

SALIDA: Áreas de visibilidad que garantizan la completa vigilancia del borde del polígono desde el exterior al mismo, cuyo número será menor o igual a  $\lfloor n/2 \rfloor$ .

Para ilustrar el algoritmo, y a modo de ejemplo, se toma un polígono no convexo de 16 vértices.

Paso 1. Dibujar el polígono  $P$ , nombrando sus vértices y aristas (Véase Figura 5.2).

El polígono  $P$  es representado a través de las coordenadas cartesianas de sus vértices. Los vértices se nombran consecutivamente siguiendo el sentido horario  $v_1, v_2, \dots, v_{n-1}, v_n$ , siendo  $n$  el número de vértices del polígono. El primer vértice,  $v_1$ , es el vértice con mayor ordenada, y a igualdad de ordenada, el de menor abscisa.

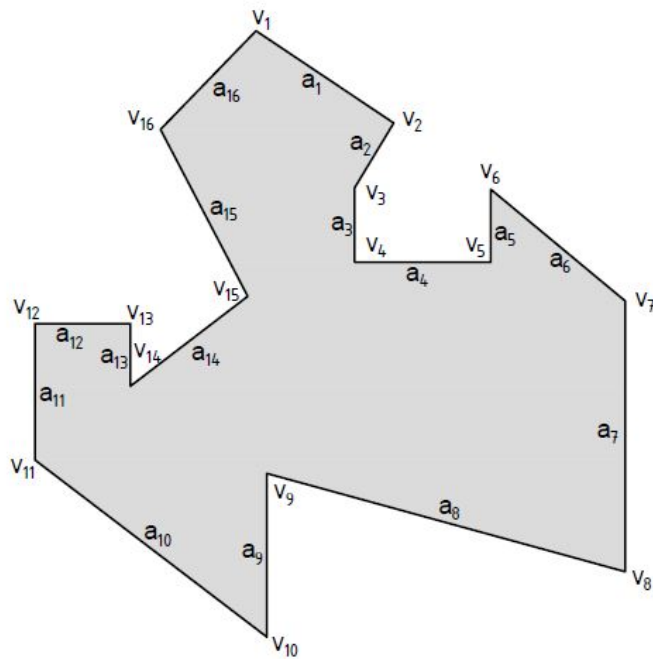


Figura 5.2: Polígono  $P$  con vértices y aristas nombradas. Elaboración Propia.

Cada arista del polígono  $\langle v_i, v_j \rangle$  se nombra, por comodidad, con el subíndice del primero de sus vértices,  $i$ . Se nombra como arista  $a_1$  a la arista adyacente al vértice  $v_1$ , en sentido horario, es decir, la arista que une  $v_1$  con  $v_2$ .

Paso 2. Construir el **Cierre Convexo** de  $P$  (Véase Figura 5.3).

Paso 3. Triangular las Bolsas. (Véase Figura 5.4).

Para triangular, se trabaja con cada bolsa de forma individual. Es decir, se trata a cada una de las bolsas como polígonos independientes. Es sabido que para triangular un polígono de  $n$  lados, hacen falta  $n - 3$  diagonales internas.

- a) Comenzar por el vértice de mayor ordenada y menor abscisa, y trazar el mayor número de diagonales internas, mediante la unión de tal vértice con el primero que se encuentra al recorrer la bolsa en sentido horario.
- b) Pasar al siguiente vértice que se encuentre en sentido horario cuando no sea posible trazar más diagonales internas, o bien si no es posible trazar ninguna diagonal desde tal vértice.

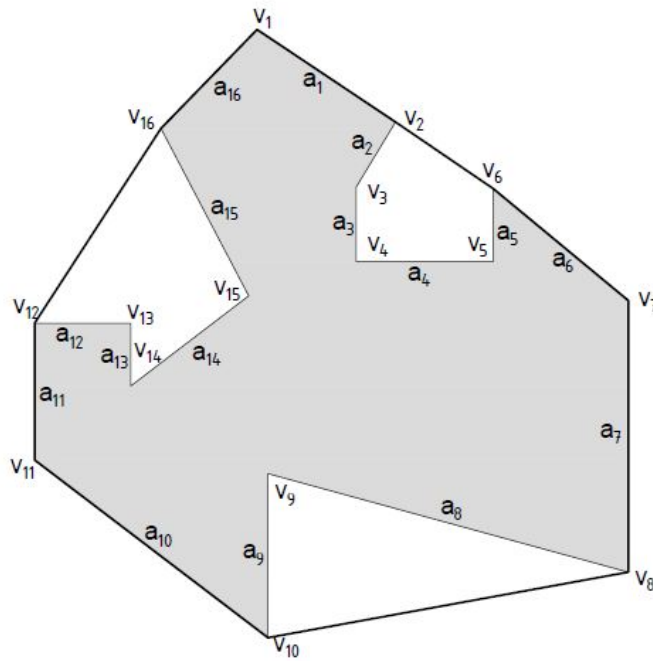


Figura 5.3: Cierre convexo. Elaboración Propia.

- c) Una vez obtenidas  $n - 3$  diagonales, concluir la triangulación de la bolsa.
- d) Se pasa a la siguiente bolsa y se repite el proceso.

Paso 4. Añadir un vértice externo al polígono  $P$ ,  $v_{\infty}$ , y hacerlo adyacente a cada vértice del cierre convexo (Véase Figura 5.5).

Con este paso se simula la triangulación de la cara exterior de  $P$ , en el sentido de hacer una partición de dicha cara en regiones cerradas delimitadas por tres curvas, como se reflejó anteriormente en la Figura 4.10.

En realidad no es una triangulación en el plano real, puesto que el vértice  $v_{\infty}$  se encuentra en un punto impropio, exterior al polígono, como se explicó en el Algoritmo 4.1.2.3.

Paso 5. Obtener  $P'$  (Véase Figura 5.6).

Este paso se divide en tres:

- a) Duplicar un vértice  $v_i$ , perteneciente al cierre convexo de  $P$ ,

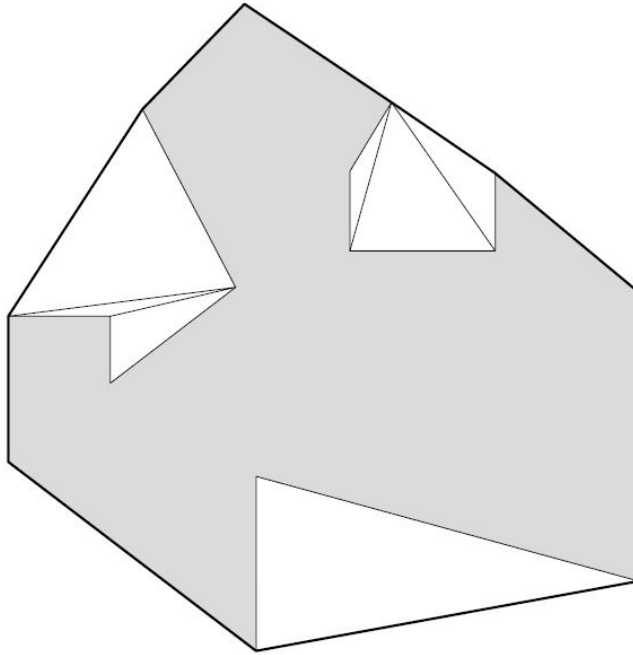


Figura 5.4: Triangulación de las bolsas. Elaboración Propia.

creándose así el vértice  $v_{i'}$  situado suficientemente próximo sobre el borde de  $P$ , en sentido horario.

- b) Hacer a  $v_{i'}$  adyacente a  $v_{\infty}$ .
- c) Abrir  $P$  entre  $v_i$  y  $v_{i'}$ . De esta forma el interior del polígono  $P$  se convierte en el exterior de un nuevo polígono  $P'$ , el cual incluye al vértice impropio  $v_{\infty}$  y al nuevo vértice  $v_{i'}$ .

Paso 6. 2-colorear los vértices pertenecientes al cierre convexo de  $P$  más el vértice  $v_{i'}$ , comenzando por dar el primer color al vértice  $v_{i'}$  y continuando con la coloración en sentido horario (Véase Figura 5.7).

Paso 7. Completar la 3-coloración de los vértices. (Véase Figura 5.8).

Tras haber 2-coloreado el polígono, solamente en los vértices que coinciden con el cierre convexo y  $v_{i'}$ , se procede a completar la 3-coloración con el resto de los vértices, teniendo en cuenta los triángulos formados en las bolsas.

Paso 8. Elegir un subconjunto de vértices del mismo color que garantice la vigilancia del exterior de  $P$ .



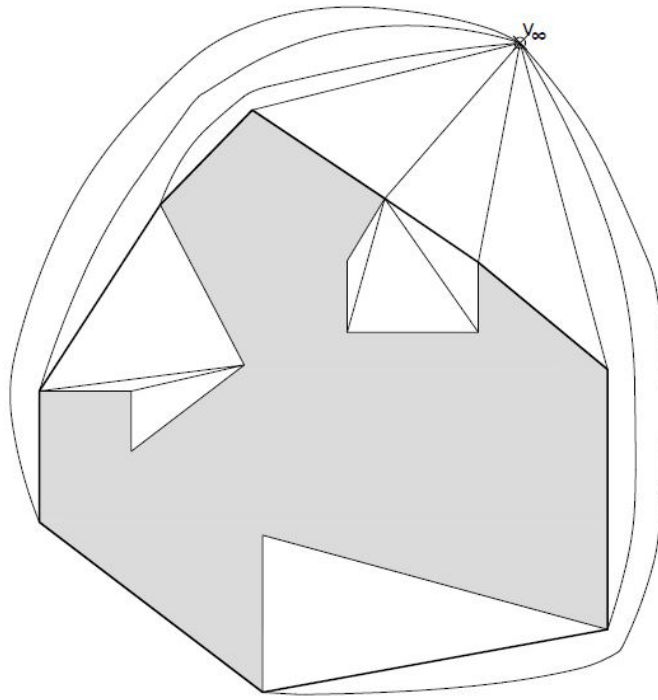


Figura 5.5: Nuevo vértice  $v_{\infty}$  y triangulación del exterior de  $P$  mediante la unión de cada vértice del cierre convexo con  $v_{\infty}$ . Elaboración Propia.

Tras el paso anterior se obtienen tres subconjuntos de vértices, teniendo en cuenta el color que le ha sido asignado a cada uno de ellos: Subconjunto de vértices de *color* A, subconjunto de vértices de *color* B, y subconjunto de vértices de *color* C. Suponiendo que el vértice  $v_{\infty}$  pertenece al subconjunto de *color* C, se ha de notar que el exterior del polígono original  $P$  es vigilado al colocar los guardias o bien, sobre el subconjunto de vértices *color* A, o bien, sobre el subconjunto de vértices *color* B. Esto es debido a que ubicando guardias en el subconjunto de *color* C no se garantiza la completa visibilidad, pues el vértice  $v_{\infty}$  no es un vértice real.

Paso 9. Obtener las áreas de visibilidad de  $P$  para un determinado subconjunto de vértices de un color.

Una vez seleccionado el subconjunto del mismo color para el cual se van a hallar las áreas de visibilidad, se comienza con el primer vértice de dicho color que se encuentra, partiendo de  $v_1$  y recorriendo los vértices de  $P$  en sentido horario. Una vez hallada el área de visibilidad de

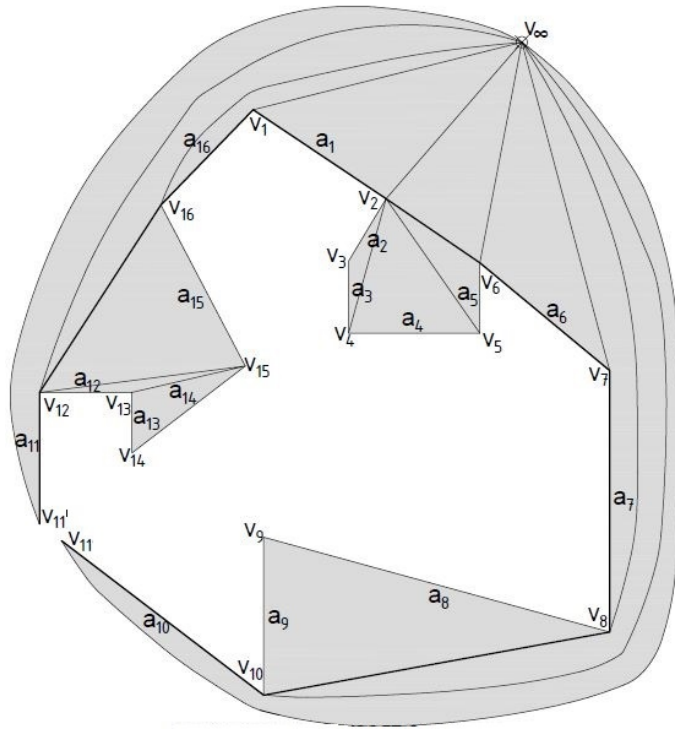


Figura 5.6: Apertura de  $P$  en un vértice  $v_i$  perteneciente al cierre convexo, creándose  $v_{i'}$ , y unión de este nuevo vértice con  $v_{\infty}$ . El interior del polígono  $P$  es ahora el exterior de polígono  $P'$ , el cual aparece sombreado. Elaboración Propia.

este primer vértice, se continúa con el siguiente hasta haber obtenido las áreas de visibilidad de todos los vértices del color seleccionado.

Tras la 3-coloración realizada en el Paso 7, cada vértice  $v_i$  tiene por misión vigilar las aristas del polígono  $P$  que definan un triángulo con el vértice  $v_i$ .

Para hallar el área de visibilidad de un vértice  $v_i$ , se ha de encontrar el área de visibilidad de cada arista,  $a_i$ , que tiene por misión vigilar, como se detalla en el siguiente procedimiento:

- a) Hallar el área de visibilidad de la arista  $a_i$ .
  - 1) Trazar la recta  $R_i$  que define la dirección de la arista  $a_i$ .
  - 2) Considerar un entorno del punto medio de la arista  $a_i$ , es decir, un disco centrado en dicho punto, de radio suficientemente pequeño. Se considera como parte del área de visibilidad el semiplano  $\alpha_{R_i}$  donde los puntos del entorno no son

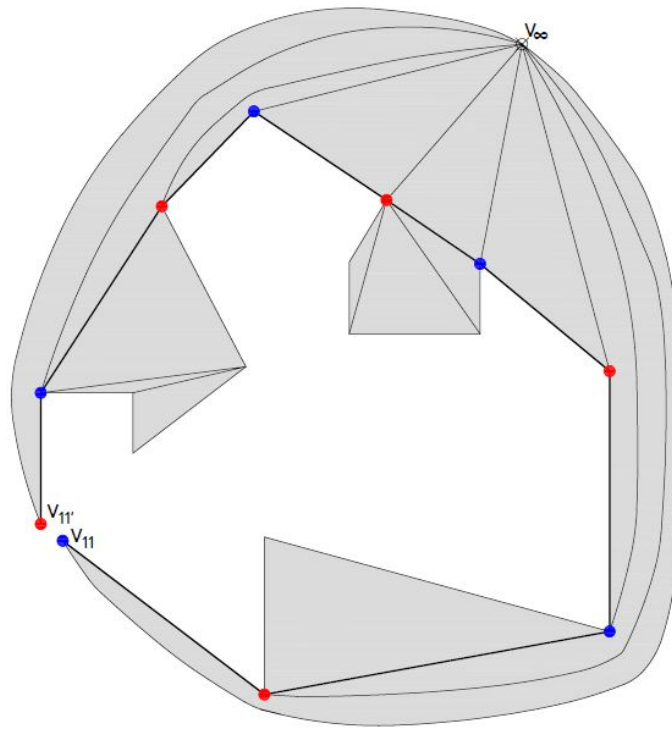


Figura 5.7: Bicoloración de los vértices del cierre convexo y el vértice  $v_{i'}$ , comenzando por éste vértice y continuando en sentido horario. Elaboración Propia.

interiores al polígono  $P$ . Por comodidad a este semiplano se le llama en adelante *semiplano exterior local* de la arista  $a_i$ .

- 3) Recorrer la recta  $R_i$  partiendo del punto medio de la arista  $a_i$ , y hacia ambos sentidos de la misma, hasta hallar el primer punto, en cada sentido, de intersección entre  $R_i$  y el borde del polígono  $P$ .

$a'$  En caso de que la recta  $R_i$  pueda ser recorrida sin que cruce con el borde del polígono, el área de visibilidad de dicha arista coincide con el semiplano  $\alpha_{R_i}$ .

$b'$  En caso de que al recorrer la dirección de  $R_i$  partiendo del punto medio de  $a_i$  en un sentido sea hallado un punto de intersección entre la recta  $R_i$  y el borde del polígono, se ha de hallar el área de visibilidad de la arista  $a_j$  del polígono con la que la recta  $R_i$  se ha cruzado.

- Repetir el Paso 9a para hallar el área de visibilidad de la arista  $a_j$  a la que pertenece el punto de intersección

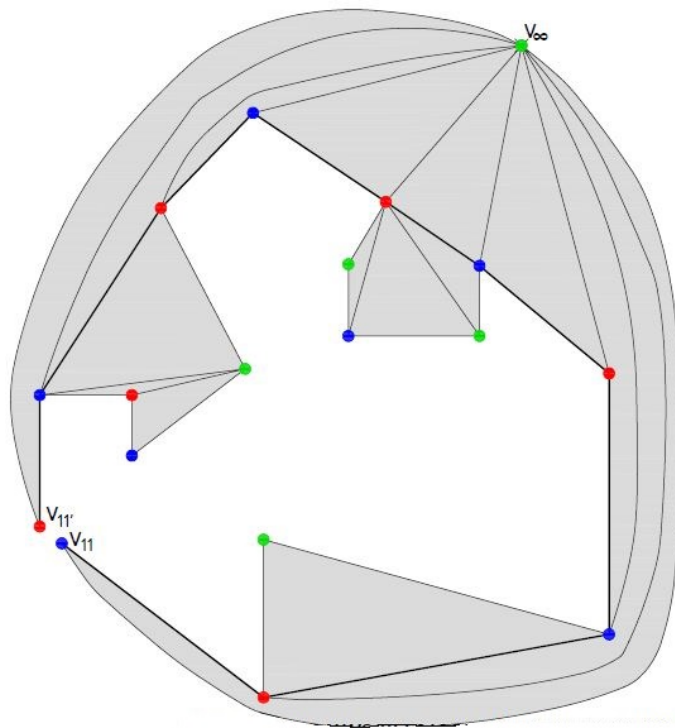


Figura 5.8: 3-coloración de los vértices y triangulación del espacio. Elaboración Propia.

hallado, considerando la arista  $a_j$ .

- Finalizar el proceso en el caso de que el punto de intersección encontrado pertenezca a una arista  $a_i$  de la cual ya se haya obtenido el semiplano  $\alpha_{R_i}$  que define a tal arista.
- 4) Hallar la intersección entre todos aquellos semiplanos considerados anteriormente, determinando así el área de visibilidad de la arista  $a_i$ .
- b) Repetir el Paso 9a para el resto de aristas que el vértice escogido tiene por misión vigilar.
  - c) Hallar la intersección de las áreas de visibilidad obtenidas para cada arista que tiene por misión vigilar para determinar el área de visibilidad del vértice elegido.
  - d) Repetir el Paso 9a para el siguiente vértice del mismo color encontrado al recorrer el polígono en sentido horario.
  - e) Finalizar el procedimiento cuando se hayan obtenido las áreas de

## 5.1. MÉTODO DETERMINISTA.

visibilidad de todos los vértices del mismo color.

Para ilustrar el procedimiento expuesto, se aplica sobre el polígono que hasta ahora se ha estado utilizando como ejemplo.

El color del subconjunto elegido es el rojo. Considerando el polígono original, en la Figura 5.9 aparecen coloreados tan sólo los vértices rojos. Cada vértice tiene por misión vigilar a las aristas que comparten triángulo, tras la triangulación realizada en el Paso 7, reflejada en la Figura 5.8.

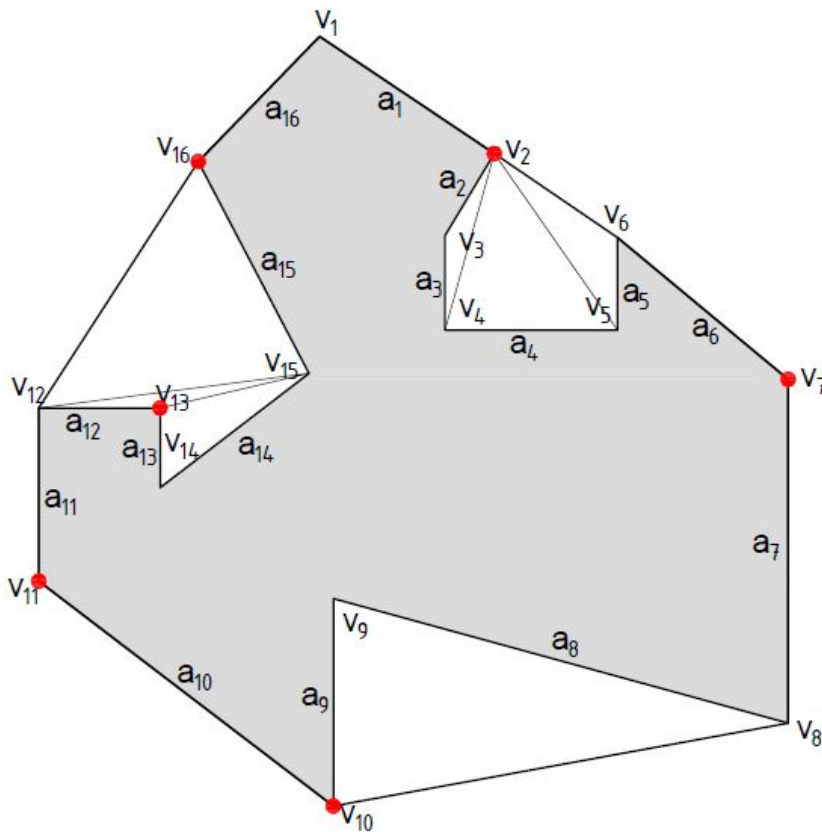


Figura 5.9: Polígono  $P$  con cierre convexo, y triangulación realizada, apreciándose los vértices de color rojo y las aristas que cada uno tiene por misión vigilar. Elaboración Propia.

- Recorrer el polígono en sentido horario, partiendo desde  $v_1$ . El primer vértice perteneciente al subconjunto de vértices de color rojo es el vértice  $v_2$ .

- Hallar el área de visibilidad del vértice  $v_2$ , el cual tiene por misión vigilar a las aristas  $a_1, a_2, a_3, a_4$  y  $a_5$ .
  - Hallar el área de visibilidad de  $a_1$  (Véase Figura 5.10).
    - ◊ Trazar la recta  $R_1$  que define la dirección de la arista  $a_1$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_1}$  de la arista  $a_1$ .
    - ◊ Recorrer la recta  $R_1$  desde el punto medio de la arista, en ambos sentidos, sin que se produzca cruce con el borde del polígono.
    - ◊ Determinar el área de visibilidad de  $a_1$  como el semiplano  $\alpha_{r_1}$ .

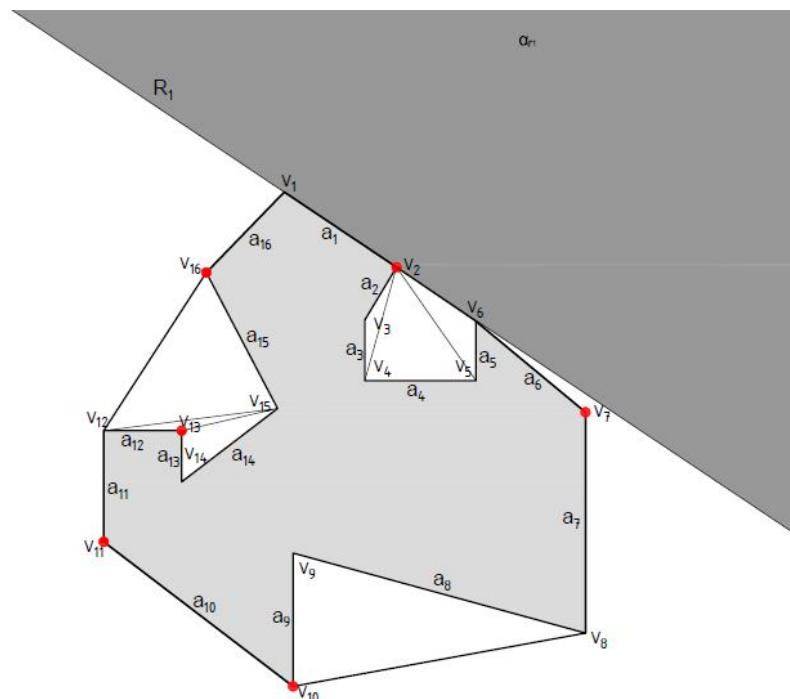


Figura 5.10: Área de visibilidad de la arista  $a_1$ , a través del trazado de la recta  $R_1$  y obtención del semiplano  $\alpha_{r_1}$ . Elaboración Propia.

- Hallar el área de visibilidad de  $a_2$  (Véase Figura 5.11).
  - ◊ Trazar la recta  $R_2$  que define la dirección de la arista  $a_2$ .
  - ◊ Hallar el semiplano exterior local  $\alpha_{r_2}$  de la arista  $a_2$ .
  - ◊ Recorrer la recta  $R_2$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_3$ .

## 5.1. MÉTODO DETERMINISTA.

- ◊ Trazar la recta  $R_3$  que define la dirección de la arista  $a_3$ .
- ◊ Hallar el semiplano exterior local  $\alpha_{r_3}$  de la arista  $a_3$ .
- ◊ Recorrer la recta  $R_3$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono en dos puntos, siendo estos puntos de intersección el vértice  $v_3$ , y el vértice  $v_4$ .
- ◊ Trazar la recta  $R_4$  que define la dirección de la arista  $a_4$ .
- ◊ Hallar el semiplano exterior local  $\alpha_{r_4}$  de la arista  $a_4$ .
- ◊ Recorrer la recta  $R_4$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono en dos puntos, siendo estos puntos de intersección el vértice  $v_4$ , y el vértice  $v_5$ .
- ◊ Trazar la recta  $R_5$  que define la dirección de la arista  $a_5$ .
- ◊ Hallar el semiplano exterior local  $\alpha_{r_5}$  de la arista  $a_5$ .
- ◊ Recorrer la recta  $R_5$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono en un punto, siendo este punto de intersección el vértice  $v_5$ .
- ◊ El área de visibilidad de  $a_2$  coincide con la intersección de los semiplanos  $\alpha_{r_2}$ ,  $\alpha_{r_3}$ ,  $\alpha_{r_4}$  y  $\alpha_{r_5}$ .

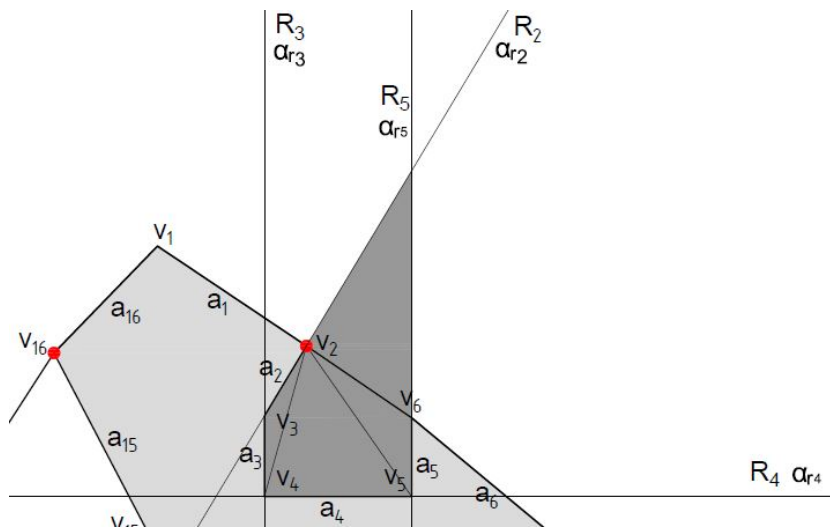


Figura 5.11: Área de visibilidad de la arista  $a_2$ . Elaboración Propia.

- ◊ Hallar el área de visibilidad de la arista  $a_3$ . Coincide con el área de visibilidad de la arista  $a_2$ .

- Hallar el área de visibilidad de la arista  $a_4$ . Coincide con el área de visibilidad de las aristas  $a_2$  y  $a_3$ .
- Hallar el área de visibilidad de la arista  $a_5$ . Coincide con el área de visibilidad de las aristas  $a_2$ ,  $a_3$  y  $a_4$ .
- Determinar la intersección de las áreas de visibilidad de  $a_1, a_2, a_3, a_4$  y  $a_5$  para hallar el área de visibilidad del vértice  $v_2$  (Véase Figura 5.12).

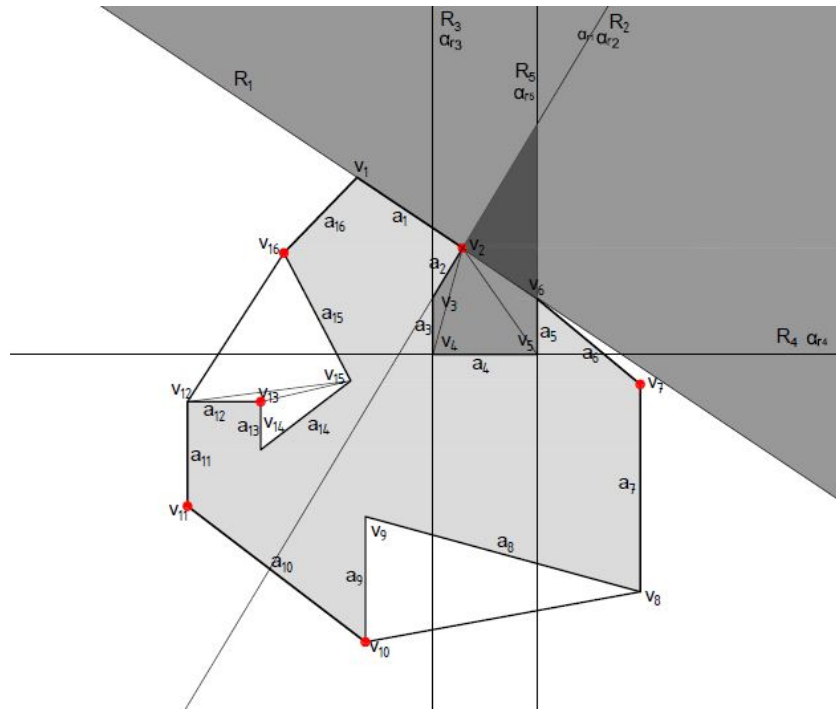


Figura 5.12: Intersección de áreas de visibilidad de las aristas  $a_1, a_2, a_3, a_4$  y  $a_5$  para la obtención del área de visibilidad de  $v_2$ , la cual se refleja más sombreada. Elaboración Propia.

- Continuar recorriendo el polígono en sentido horario para encontrar el siguiente vértice de color rojo, siendo éste  $v_7$ .
- Hallar el área de visibilidad del vértice  $v_7$ , el cual tiene por misión vigilar a las aristas  $a_6$  y  $a_7$ .
  - Hallar el área de visibilidad de la arista  $a_6$ .
    - ◊ Trazar la recta  $R_6$  que define la dirección de la arista  $a_6$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_6}$  de la arista  $a_6$ .



- ◊ Recorrer la recta  $R_6$  desde el punto medio de la arista, en ambos sentidos, sin que se produzca el cruce con el borde del polígono.
    - ◊ Determinar el área de visibilidad de  $a_6$  como el semiplano  $\alpha_{r_6}$ .
  - Hallar el área de visibilidad de la arista  $a_7$ .
    - ◊ Trazar la recta  $R_7$  que define la dirección de la arista  $a_7$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_7}$  de la arista  $a_7$ .
    - ◊ Recorrer la recta  $R_7$  desde el punto medio de la arista, en ambos sentidos, sin que se produzca el cruce con el borde del polígono.
    - ◊ Determinar el área de visibilidad de  $a_7$  como el semiplano  $\alpha_{r_7}$ .
- Determinar la intersección de las áreas de visibilidad de  $a_6$  y  $a_7$  para hallar el área de visibilidad del vértice  $v_7$  (Véase Figura 5.13).
- Continuar recorriendo el polígono en sentido horario para encontrar el siguiente vértice de color rojo, siendo éste  $v_{10}$ .
- Hallar el área de visibilidad del vértice  $v_{10}$ , el cual tiene por misión vigilar a las aristas  $a_8, a_9$  y  $a_{10}$ .
  - Hallar el área de visibilidad de  $a_8$ .
    - ◊ Trazar la recta  $R_8$  que define la dirección de la arista  $a_8$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_8}$  de la arista  $a_8$ .
    - ◊ Recorrer la recta  $R_8$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_9$ .
    - ◊ Trazar la recta  $R_9$  que define la dirección de la arista  $a_9$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_9}$  de la arista  $a_9$ .
    - ◊ Recorrer la recta  $R_9$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_8$ .
    - ◊ El área de visibilidad de  $a_8$  coincide con la intersección de los semiplanos  $\alpha_{r_8}$  y  $\alpha_{r_9}$ .
  - Hallar el área de visibilidad de  $a_9$ . Coincide con el área de visibilidad de  $a_8$ .
  - Hallar el área de visibilidad de  $a_{10}$ .
    - ◊ Trazar la recta  $R_{10}$  que define la dirección de la arista  $a_{10}$ .

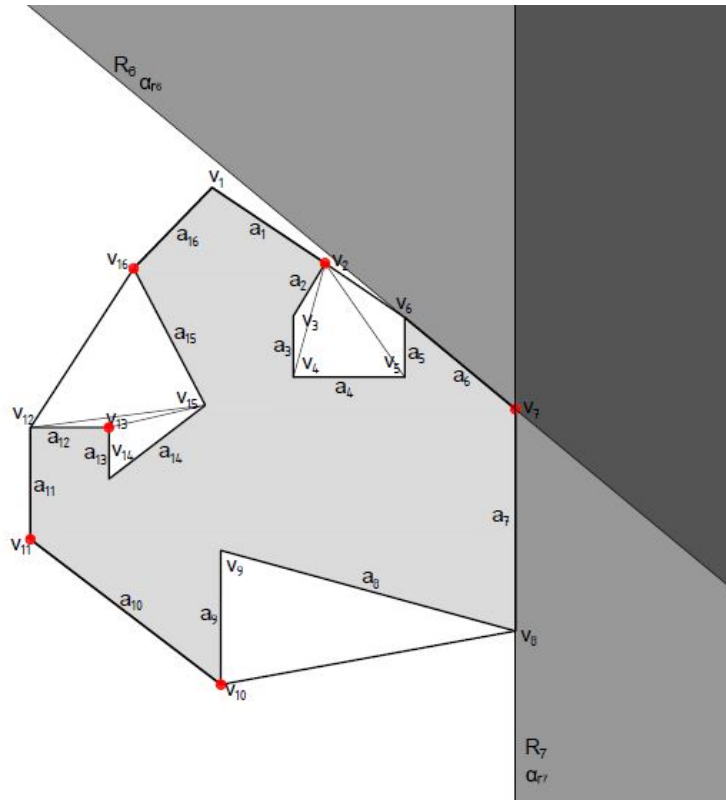


Figura 5.13: Intersección de áreas de visibilidad de las aristas  $a_6$  y  $a_7$  para la obtención del área de visibilidad de  $v_7$ , la cual se refleja más sombreada. Elaboración Propia.

- ◊ Hallar el semiplano exterior local  $\alpha_{r_{10}}$  de la arista  $a_{10}$ .
  - ◊ Recorrer la recta  $R_{10}$  desde el punto medio de la arista, en ambos sentidos, sin que se produzca cruce con el borde del polígono.
  - ◊ Determinar el área de visibilidad de  $a_{10}$  como el semiplano  $\alpha_{r_{10}}$ .
- Determinar la intersección de las áreas de visibilidad de  $a_8, a_9$  y  $a_{10}$  para hallar el área de visibilidad del vértice  $v_{10}$  (Véase Figura 5.14).
- Continuar recorriendo el polígono en sentido horario para encontrar el siguiente vértice de color rojo, siendo éste  $v_{11}$ .
- Hallar el área de visibilidad del vértice  $v_{11}$ , el cual tiene por misión vigilar únicamente a la arista  $a_{11}$ .

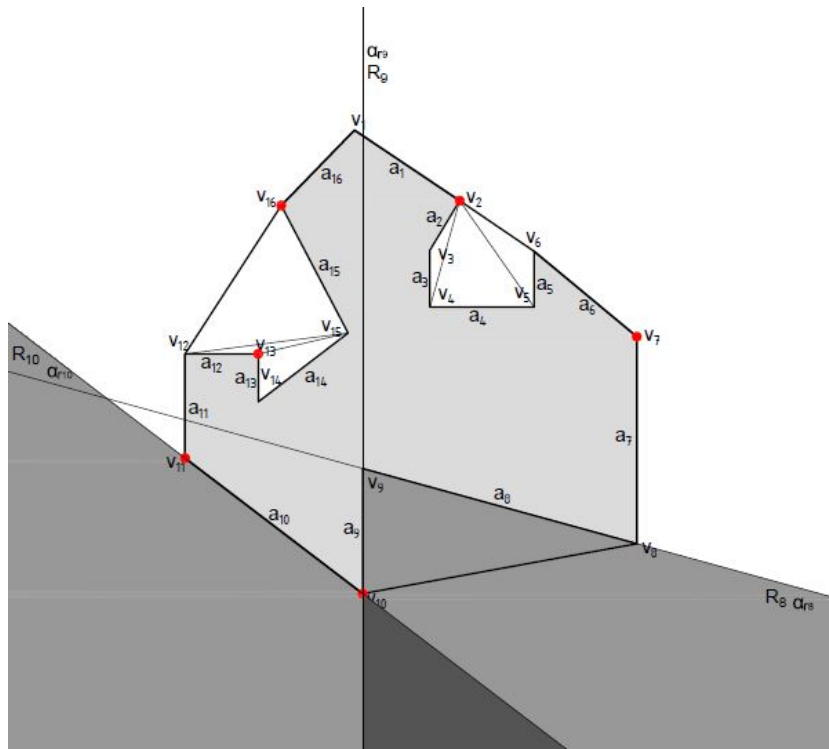


Figura 5.14: Intersección de áreas de visibilidad de las aristas  $a_8, a_9$  y  $a_{10}$  para la obtención del área de visibilidad de  $v_{10}$ , la cual se refleja más sombreada. Elaboración Propia.

- Hallar el área de visibilidad de  $a_{11}$ .
  - ◊ Trazar la recta  $R_{11}$  que define la dirección de la arista  $a_{11}$ .
  - ◊ Hallar el semiplano exterior local  $\alpha_{r_{11}}$  de la arista  $a_{11}$ .
  - ◊ Recorrer la recta  $R_{11}$  desde el punto medio de la arista, en ambos sentidos, sin que se produzca el cruce con el borde del polígono.
  - ◊ Determinar el área de visibilidad de  $a_{11}$  como el semiplano  $\alpha_{r_{11}}$ .
- Determinar el área de visibilidad del vértice  $v_{11}$  como el área de visibilidad de  $a_{11}$  (Véase Figura 5.15).
- Continuar recorriendo el polígono en sentido horario para encontrar el siguiente vértice de color rojo, siendo éste  $v_{13}$ .
  - Hallar el área de visibilidad del vértice  $v_{13}$ , el cual tiene por misión

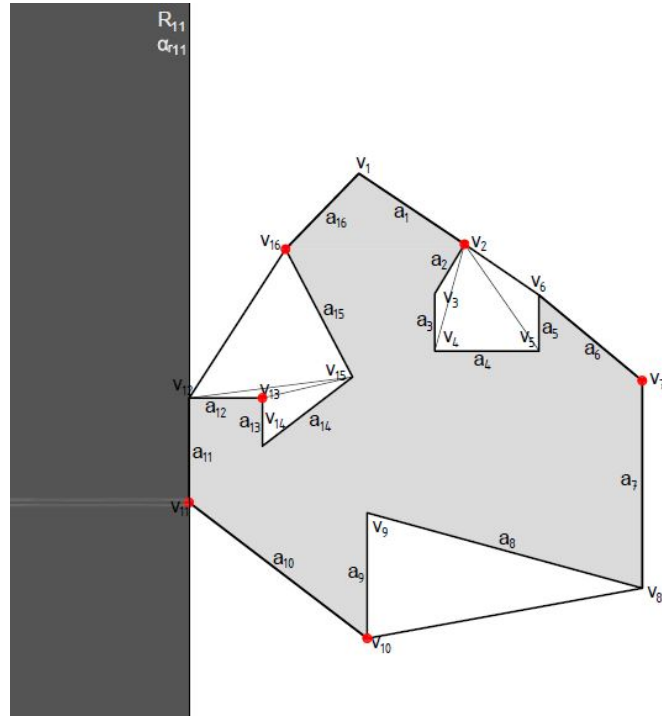


Figura 5.15: Área de visibilidad de  $v_{11}$ . Elaboración Propia.

vigilar a las aristas  $a_{12}$ ,  $a_{13}$  y  $a_{14}$ .

- Hallar el área de visibilidad de la arista  $a_{12}$ .
  - ◊ Trazar la recta  $R_{12}$  que define la dirección de la arista  $a_{12}$ .
  - ◊ Hallar el semiplano exterior local  $\alpha_{r_{12}}$  de la arista  $a_{12}$ .
  - ◊ Recorrer la recta  $R_{12}$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección un punto perteneciente a la arista  $a_{14}$ .
  - ◊ Trazar la recta  $R_{14}$  que define la dirección de la arista  $a_{14}$ .
  - ◊ Hallar el semiplano exterior local  $\alpha_{r_{14}}$  de la arista  $a_{14}$ .
  - ◊ Recorrer la recta  $R_{14}$  desde el punto medio de la arista, en ambos sentidos, produciéndose dos cruces con el borde del polígono, siendo los puntos de intersección el vértice  $v_{14}$  y  $v_{15}$ .
  - ◊ Trazar la recta  $R_{13}$  que define la dirección de la arista  $a_{13}$ .
  - ◊ Hallar el semiplano exterior local  $\alpha_{r_{13}}$  de la arista  $a_{13}$ .

- ◊ Recorrer la recta  $R_{13}$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_{14}$ .
    - ◊ Trazar la recta  $R_{15}$  que define la dirección de la arista  $a_{15}$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_{15}}$  de la arista  $a_{15}$ .
    - ◊ Recorrer la recta  $R_{15}$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_{15}$ .
    - ◊ El área de visibilidad de  $a_{12}$  coincide con la intersección de los semiplanos  $\alpha_{r_{12}}$ ,  $\alpha_{r_{13}}$ ,  $\alpha_{r_{14}}$  y  $\alpha_{r_{15}}$ .
  - Hallar el área de visibilidad de la arista  $a_{13}$ .
    - ◊ Trazar la recta  $R_{13}$  que define la dirección de la arista  $a_{13}$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_{13}}$  de la arista  $a_{13}$ .
    - ◊ Recorrer la recta  $R_{13}$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_{14}$ .
    - ◊ Trazar la recta  $R_{14}$  que define la dirección de la arista  $a_{14}$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_{14}}$  de la arista  $a_{14}$ .
    - ◊ Recorrer la recta  $R_{14}$  desde el punto medio de la arista, en ambos sentidos, produciéndose dos cruces con el borde del polígono, siendo los puntos de intersección el vértice  $v_{14}$  y  $v_{15}$ .
    - ◊ Trazar la recta  $R_{15}$  que define la dirección de la arista  $a_{15}$ .
    - ◊ Hallar el semiplano exterior local  $\alpha_{r_{15}}$  de la arista  $a_{15}$ .
    - ◊ Recorrer la recta  $R_{15}$  desde el punto medio de la arista, en ambos sentidos, produciéndose un cruce con el borde del polígono, siendo el punto de intersección el vértice  $v_{15}$ .
    - ◊ El área de visibilidad de  $a_{13}$  coincide con la intersección de los semiplanos  $\alpha_{r_{13}}$ ,  $\alpha_{r_{14}}$  y  $\alpha_{r_{15}}$ .
  - Hallar el área de visibilidad de la arista  $a_{14}$ . Coincide con el área de visibilidad de  $a_{13}$ .
- Determinar el área de visibilidad del vértice  $v_{13}$  como la intersección entre las áreas de visibilidad de las aristas  $a_{12}$ ,  $a_{13}$  y  $a_{14}$  (Véase Figura 5.16).
- Continuar recorriendo el polígono en sentido horario para encontrar el último vértice de color rojo, siendo éste  $v_{16}$ .
  - Hallar el área de visibilidad del vértice  $v_{16}$ , el cual tiene por misión vigilar a las aristas  $a_{15}$  y  $a_{16}$  (Véase Figura 5.17).

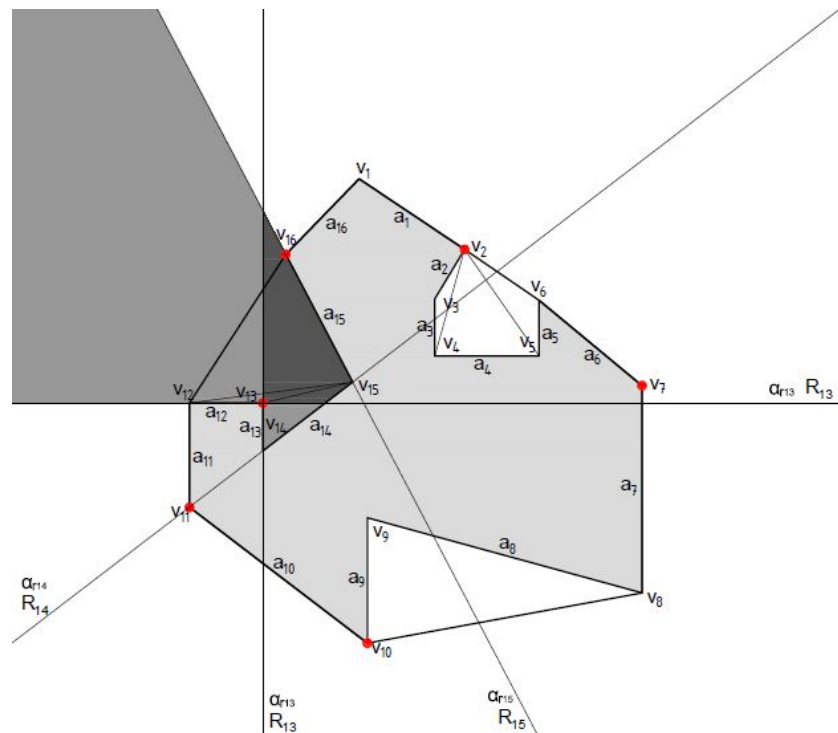


Figura 5.16: Área de visibilidad de  $v_{13}$ . Elaboración Propia.

- Finalizar el proceso de obtención de áreas de visibilidad, y estudiar posibles solapes de las áreas calculadas (Véase Figura 5.18).

En conclusión, en este apartado se ha explicado de forma detallada el algoritmo diseñado, basado en el Algoritmo de la Fortaleza, el cual, dada la planta de un edificio, es capaz de devolver las mínimas áreas de visibilidad desde las que es posible vigilar de forma completa el exterior de dicho edificio.

### 5.1.3. Descripción del Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Polígonos Ortogonales

En aquellos edificios que son susceptibles de ser representados por plantas con forma de polígono ortogonal, el número de guardias para vigilar el contorno del edificio será menor.

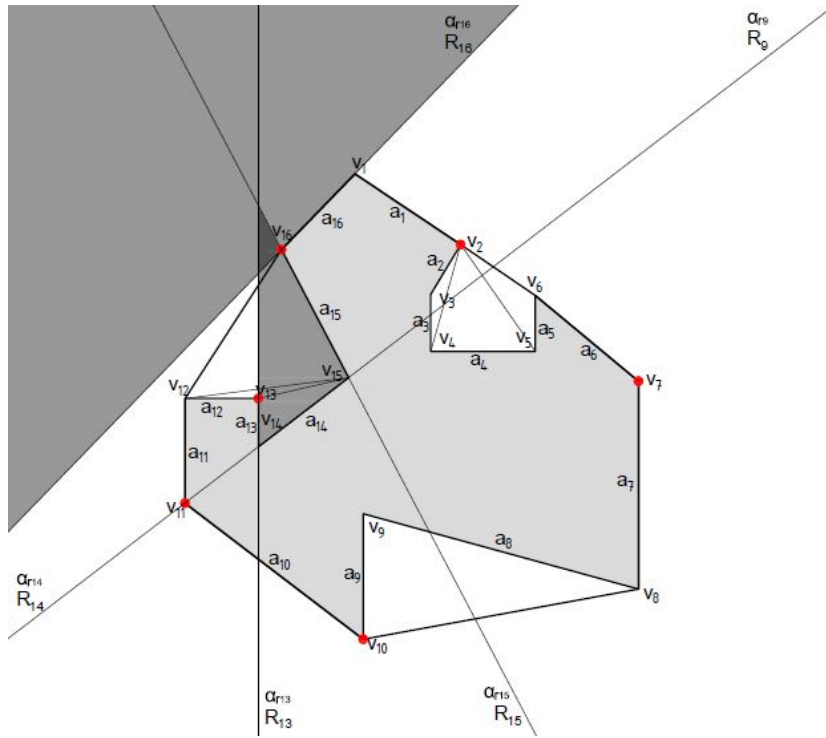


Figura 5.17: Área de visibilidad de  $v_{16}$ . Elaboración Propia.

Como ya se ha expuesto en el Apartado 4.1.2.3, aunque son necesarios más guardias para cubrir el exterior que el interior de un polígono arbitrario,  $\lceil n/2 \rceil$  para el exterior *versus*  $\lfloor n/3 \rfloor$  para el interior; con polígonos ortogonales el número requerido difiere sólo levemente,  $\lceil n/4 \rceil + 1$  para el exterior *versus*  $\lfloor n/4 \rfloor$  para el interior.

Tomando como base la Prueba del Teorema 4.1.8, (Aggarwal, 1984), por la cual se describen los pasos a seguir para obtener los guardias en vértices suficientes para vigilar el exterior de un polígono ortogonal, se realiza la adaptación del mismo para la obtención de las áreas de visibilidad a lo largo de las cuales los guardias obtenidos pueden desplazarse sin perder la visibilidad de la porción del borde de polígono que le ha sido asignada.

ENTRADA: Polígono ortogonal  $P$  de  $n$  vértices.

SALIDA: Áreas de visibilidad que garantizan la completa vigilancia del borde del polígono desde el exterior al mismo, cuyo número será menor o igual a  $\lceil n/4 \rceil + 1$ .

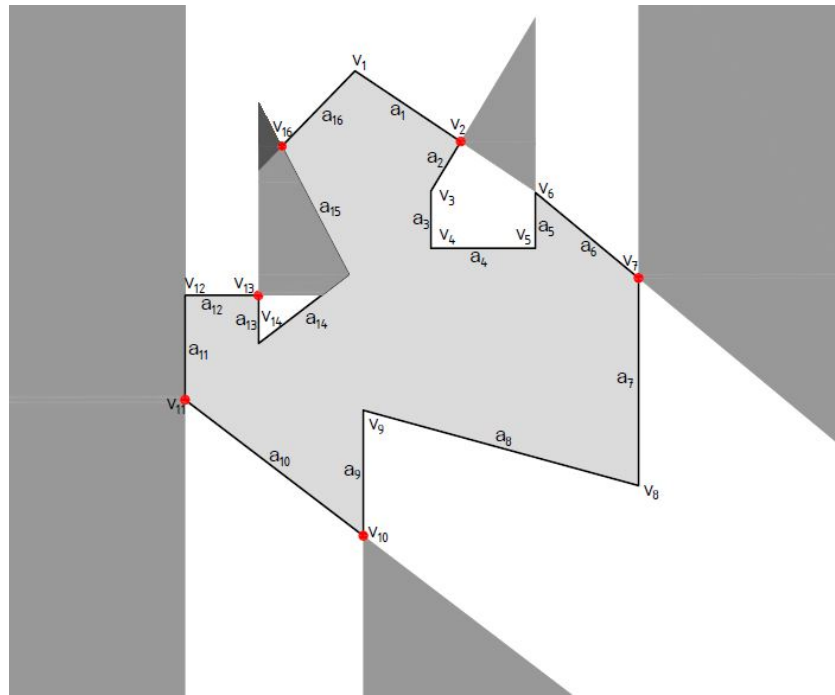


Figura 5.18: Representación de todas las áreas de visibilidad del subconjunto de vértices de color rojo. Obsérvese que la zona más sombrada es debido al solape de dos áreas de visibilidad, la de  $v_{13}$  y la de  $v_{16}$ . Elaboración Propia.

Paso 1. Dibujar el polígono  $P$  de  $n$  vértices, nombrando sus vértices y aristas (Véase Figura 5.19).

Se realiza de la misma forma que en el Paso 5.1.2 del Apartado 5.1.2.

Paso 2. Construir un recuadro tal que el polígono  $P$  quede situado en su interior. (Véase Figura 5.20).

Paso 3. Sea  $e$  la arista horizontal de  $P$  cuyos vértices poseen la mayor coordenada  $Y$  (o cualquiera con la máxima altura si es que hubiera varias), eliminar dicha arista  $e$  (Véase Figura 5.21).

Paso 4. Extender las dos aristas adyacentes verticales a  $e$  en sentido ascendente, cerrando el polígono completo con el rectángulo que lo rodea, creándose un nuevo polígono  $P'$  compuesto por  $n + 4$  vértices (Véase Figura 5.22).

El interior de  $P'$  coincide con el exterior que rodea a  $P$ , excepto por el rectángulo  $Q$ . Este nuevo polígono  $P'$  es también ortogonal.



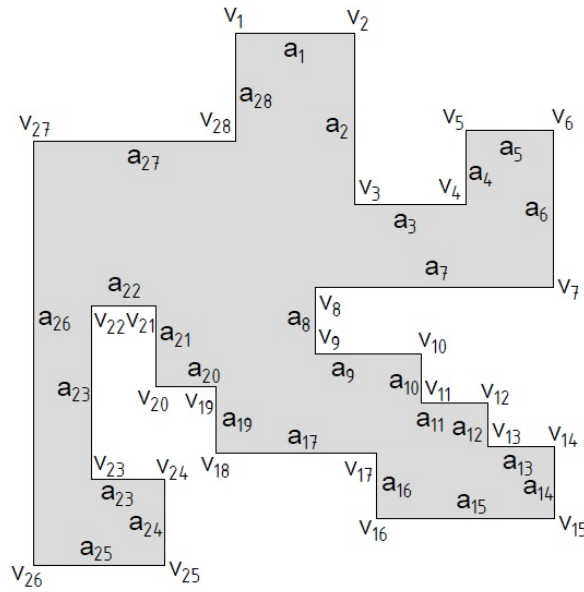


Figura 5.19: Polígono ortogonal, P. Elaboración Propia.

Paso 5. Hacer una partición de  $P'$  en cuadriláteros (Véase Figura 5.23).

Se ha de tener en cuenta que una *cuadrilaterización* debe ser convexa, es decir, no basta con unir pares de triángulos, como se expuso en el Apartado 4.1.2.2.

Paso 6. 4-colorear los vértices de  $P'$  (Véase Figura 5.24).

Para la 4-coloración se ha de elegir el orden de preferencia de asignación de colores, como  $c_1, c_2, c_3$  y  $c_4$ . Se debe tener en cuenta que en la 3-coloración una vez coloreados los dos primeros vértices, el resto de vértices se colorean de forma unívoca. Esto no ocurre en la 4-coloración, así que se debe establecer un orden de asignación de los colores.

Por ello, se debe elegir un color  $c_1$  el cual nunca quede asignado a alguno de los cuatro vértices de  $P'$  que no coinciden con los vértices del polígono original  $P$ . Ese color  $c_1$  debe colocarse también de forma que garantice la vigilancia del rectángulo  $Q$ . Por ello, ese primer color  $c_1$  se colocará sobre el vértice que sustituye a  $v_1$  de  $P$ , creado al haberse eliminado la arista  $e$  y al haber extendido la arista  $v_n$ .

- Elegir un color,  $c_1$ , y asignar al vértice creado al extender la arista

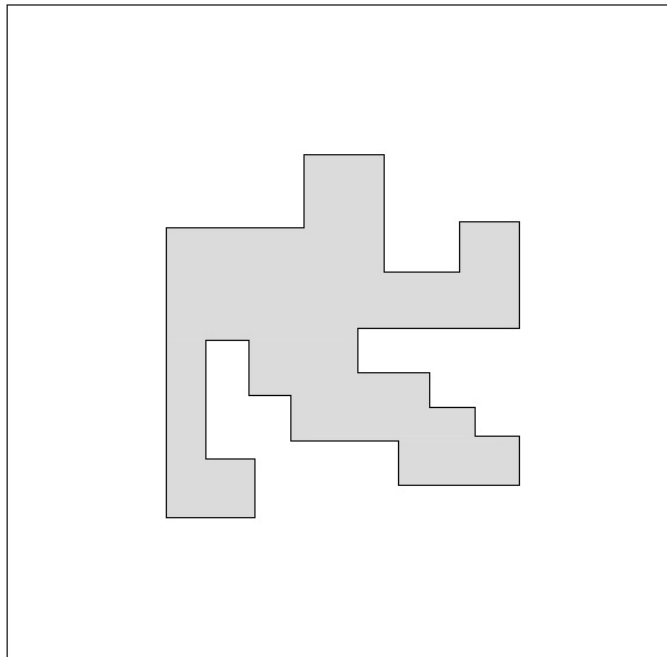


Figura 5.20: Polígono ortogonal,  $P$ , encerrado en recuadro. Elaboración Propia.

$e$ , de menor abscisa, es decir, el vértice creado que sustituye a  $v_1$ .

- Recorrer los vértices de  $P'$  en sentido horario (lo cual significa recorrer los vértices de  $P$  en sentido antihorario), coloreando los vértices del cuadrilátero creado con otros tres colores. Al siguiente vértice encontrado que pertenezca al polígono  $P$  se le asigna el color  $c_2$ , al siguiente vértice hallado que pertenezca a  $P$  se le asigna el color  $c_3$ , y por último se le asigna el color  $c_4$ .
- Una vez concluida la coloración de un cuadrilátero, continuar recorriendo los vértices de  $P'$ , para hallar el siguiente cuadrilátero para colorear.
- Colorear el siguiente vértice de  $P'$  que coincida con un vértice de  $P$ . En primer lugar, intentar colorear con el color  $c_1$ . Si no es posible, colorear con el color  $c_2$ ,  $c_3$  o  $c_4$  en ese orden de preferencia.
- Completar la 4-coloración de los cuadriláteros de la misma forma.
- Cualquier subconjunto de vértices de un color que no haya sido colocado en ninguno de los vértices pertenecientes al recuadro y que vigile a  $Q$ , garantiza la correcta vigilancia del exterior de  $P$ .

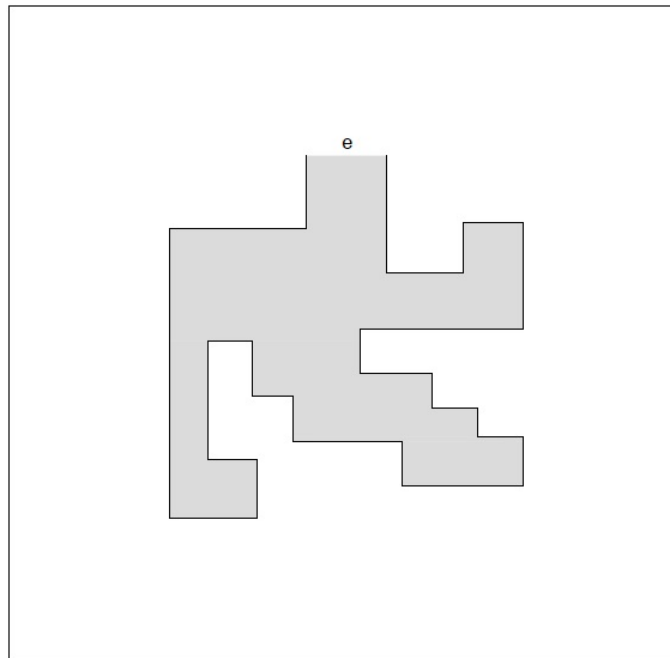


Figura 5.21: Eliminación de la arista superior  $e$ . Elaboración Propia.

Paso 7. Una vez que se han coloreado los vértices del polígono  $P'$ , volver a considerar el polígono original  $P$ , desplazando los vértices creados al eliminar la arista  $e$ , a su posición primaria.

Paso 8. Hallar las áreas de visibilidad de un subconjunto de vértices de color que garantice la vigilancia completa del borde de  $P$ .

Aquel subconjunto de vértices que compartan color, tal que nunca exista ningún vértice de ese color sobre alguno de los cuatro vértices creados en el Paso 4, y que algún vértice de dicho color vigile al rectángulo  $Q$ , es un subconjunto de vértices de color que garantiza la vigilancia completa del borde de  $P$ .

Para hallar las áreas de visibilidad se sigue el mismo procedimiento que el desarrollado en el Paso 9 del Apartado 5.1.2, con la particularidad de que en el caso de un polígono ortogonal, cada vértice seleccionado tiene por misión vigilar las aristas del polígono  $P$  que, tras la cuadrilaterización se encuentren compartiendo algún cuadrilátero con dicho vértice (Véase Figura 5.25).

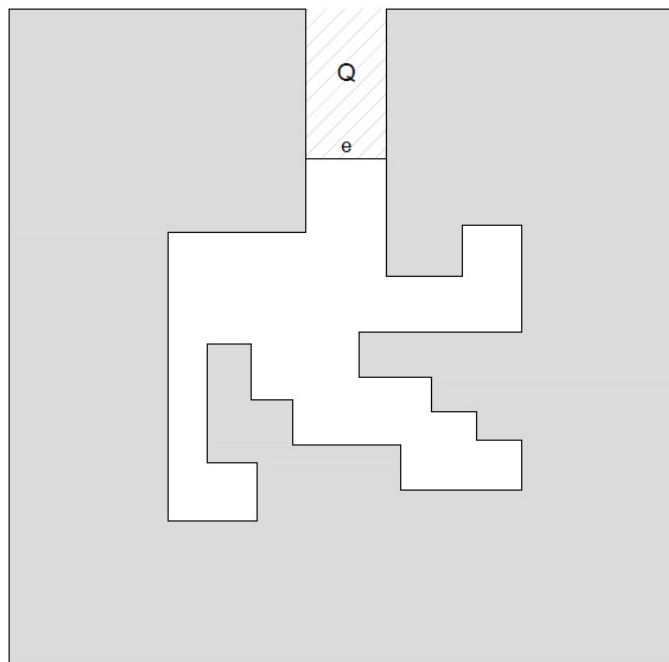


Figura 5.22: Extensión de aristas adyacentes a  $e$ , creación de polígono  $P'$  de  $n + 4$  vértices, y rectángulo  $Q$ . Elaboración Propia.

#### 5.1.4. Implementación

El algoritmo presentado se ha programado en lenguaje *PYTHON*. Este trabajo fue realizado con la colaboración del Profesor Doctor José Ramón Portillo Fernández, profesor del Departamento de Matemática Aplicada I de la Universidad de Sevilla y Félix Martín López, como Trabajo Fin de Grado en Tecnologías Informáticas de este último. Dicho trabajo tiene por nombre *Aplicación Web de Posicionamiento Topográfico* (Martín López, 2015b).

La aplicación diseñada precisa de la introducción de las coordenadas cartesianas que delimitan la forma del polígono a vigilar, así como el orden de unión entre dichos vértices. Una vez definido el polígono a vigilar, la aplicación es capaz de obtener las áreas de visibilidad suficientes para vigilar el borde del polígono que representa el recinto exterior del edificio, desde el lado exterior del mismo.

Dicha aplicación se ha diseñado para obtener resultados para cualquier tipo de polígono, siguiendo los pasos descritos en el Apartado 5.1.2.

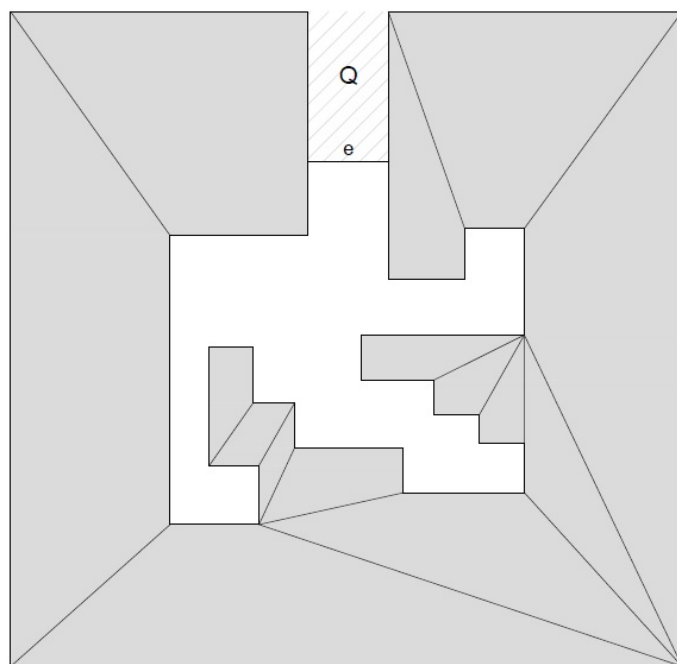


Figura 5.23: Cuadrilaterización de P'. Elaboración Propia.

La interfaz de esta aplicación web es sencilla como puede apreciarse en la Figura 5.26.

La plataforma que se ha utilizado es *Eclipse Kepler Service Release 1*, y los *pluggins* necesarios han sido *PyDev* y *Aptana Studio 3.4.2*.

Las librerías y módulos utilizados han sido los siguientes:

- Librería *D3*. Con ella se consigue representar la planta del edificio y las zonas de visión en la aplicación web, controlado desde *JavaScript*.
- Librería *Graphics*. Tiene la misma finalidad que la anterior, con la diferencia de que las representaciones se realizan en la plataforma de programación.
- Librería *Shapely*. Muy útil para el cálculo de las áreas de visibilidad, ya que consigue solucionar de forma sencilla el problema de las intersecciones entre polígonos.
- Librería *jQuery*.

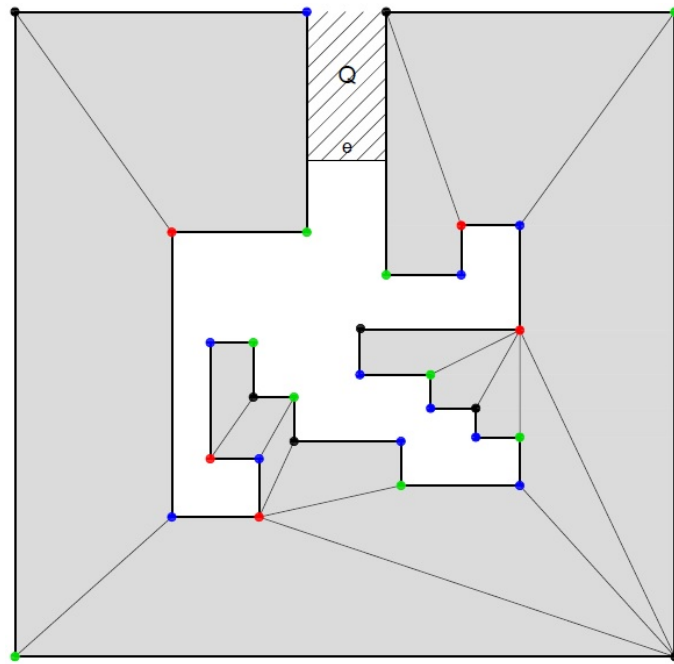


Figura 5.24: 4-Coloración de los vértices de  $P'$ . Elaboración Propia.

- Módulo *Time*. Permite controlar el tiempo de ejecución del programa.
- Módulo *JSON*.

Se optó por utilizar el lenguaje de programación *PYTHON*<sup>1</sup> Para poder hacer uso de la aplicación se debe tener instalado en el equipo el *software Python*.

### 5.1.5. Resultados

Los resultados obtenidos en esta sección son tres:

<sup>1</sup>Este lenguaje está teniendo una gran repercusión en los últimos años, estableciéndose como el lenguaje favorito de muchos programadores. En 2011 Linux le otorgó el premio al mejor lenguaje de programación por votación de desarrolladores y usuarios. Dicho lenguaje es un lenguaje multiparadigma, es decir, permite al programador optar por varios estilos de programación en lugar de forzarle a uno particular. Es administrado por la *Python Software Foundation* y fue creado a finales de los ochenta por Guido van Rossum en el Centro para las Matemáticas y la Informática en los Países Bajos. Fue diseñado con la finalidad de ser leído con facilidad, utilizando palabras, en lugar de símbolos.

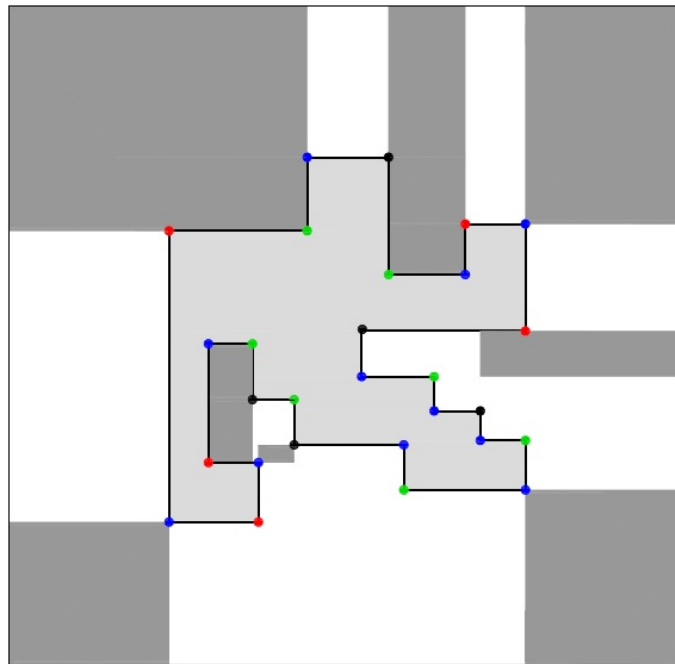


Figura 5.25: Áreas de visibilidad obtenidas para garantizar la vigilancia de  $P$ . Elaboración Propia.

- Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Caso General.
- Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo: Polígonos ortogonales.
- Implementación del primero de los algoritmos, para su uso a través de una aplicación web.

El algoritmo implementado no se encuentra actualmente disponible en Internet, aunque se está trabajando en ello. No obstante, el código completo está disponible en (Martín López, 2015a) y en la documentación digital anexa a esta memoria de tesis.

Cabe destacar la versatilidad del algoritmo diseñado, ya que responde de forma adecuada ante distintos casos. Por lo que puede garantizarse que el método expuesto es aplicable para vigilar el borde de un polígono desde el exterior



Figura 5.26: Pantalla de representación de datos (Martín López, 2015b).

del mismo.

En las Figuras 5.27, 5.28, 5.29, 5.30, 5.31 y 5.32 se muestran diferentes polígonos sobre los cuales se ha aplicado el algoritmo desarrollado en el Apartado 5.1.2 utilizando la implementación descrita en el Apartado 5.1.4.

En todas estas figuras puede verificarse que el número de áreas de visibilidad obtenidas no excede a  $\lfloor n/2 \rfloor$ , siendo  $n$  el número de vértices del polígono a vigilar.

### 5.1.6. Conclusiones

Tras analizar este primer método diseñado a partir de un algoritmo determinista, y de su implementación, se concluye que el algoritmo presentado permite conocer de forma sencilla un número suficiente de áreas de visibilidad que garantizan la completa visión de un edificio desde el exterior del mismo.

El algoritmo responde adecuadamente independientemente de la dificultad de los datos de entrada. Además, se ejecuta rápidamente. Ciertamente esta aplicación facilita el trabajo de toma de decisión de qué zona elegir para realizar una vigilancia óptima de un objeto, como ha podido observarse en las figuras expuestas anteriormente, en las que polígonos con formas complejas



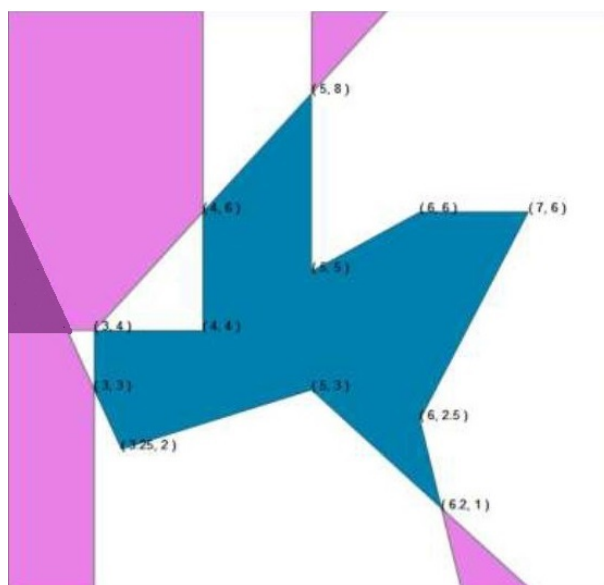


Figura 5.27: Prueba sobre polígono de forma simple (Martín López, 2015b).

pueden ser vigilados desde un número significativamente menor de posicionamientos a los que a primera vista se hubiera podido pensar.

Además, es importante destacar que en las referencias consultadas en relación a los Problemas de Vigilancia, no se había propuesto un algoritmo de vigilancia para este caso concreto: la vigilancia del borde de un polígono desde el exterior del mismo.

Ahora bien, como en muchos de los algoritmos desarrollados en la Sección 4.1, éste no tiene en cuenta la limitación visual del elemento que vigila. Es decir, esta propuesta es útil siempre y cuando el vigilante no tenga un alcance limitado. Tampoco se tiene en cuenta en este algoritmo el ángulo de incidencia del rayo de visión sobre el paramento a vigilar.

Aún así, cabe destacar la gran utilidad de este método para problemas de vigilancia, y su posible aplicación a trabajos de inspección y cobertura no necesariamente vinculados al área de la Topografía.

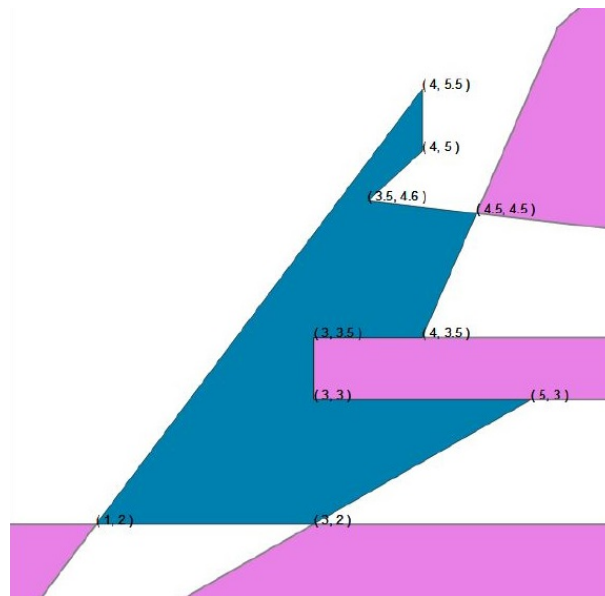


Figura 5.28: Prueba sobre polígono de forma compleja (Martín López, 2015b).



Figura 5.29: Prueba sobre polígono de forma compleja 2 (Martín López, 2015b).

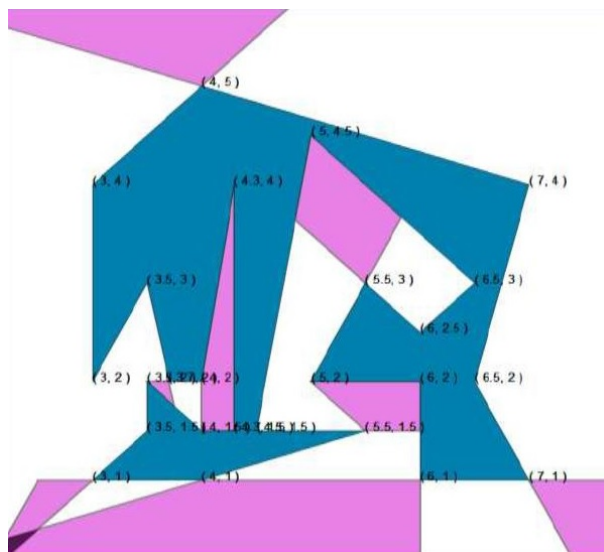


Figura 5.30: Prueba sobre polígono de forma compleja 3 (Martín López, 2015b).

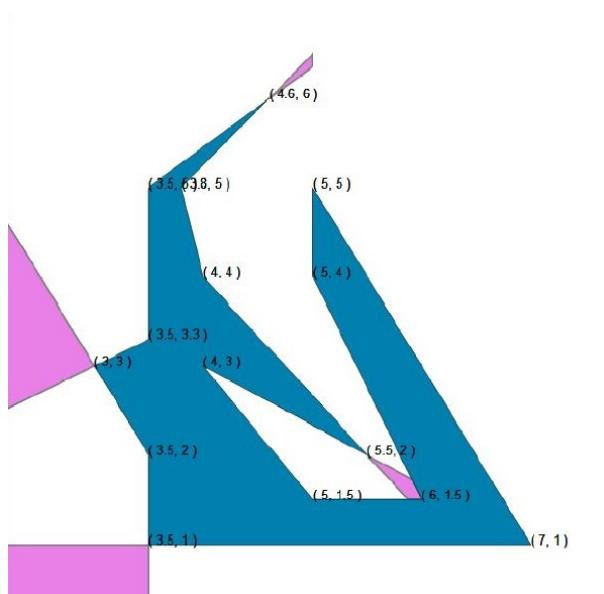


Figura 5.31: Prueba sobre polígono de forma compleja 4 (Martín López, 2015b).

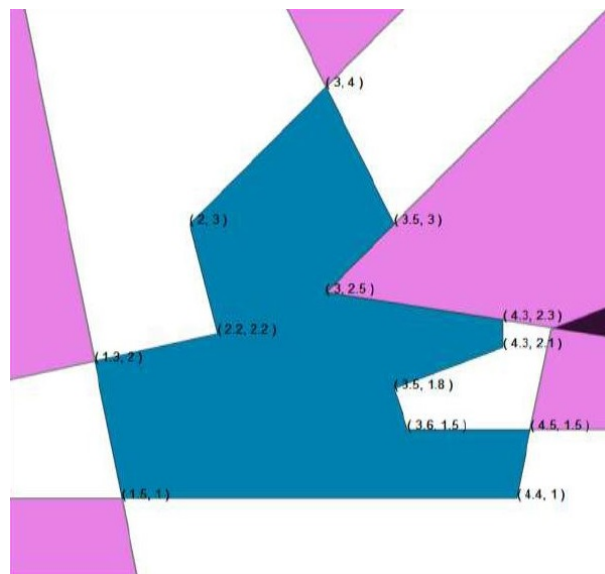


Figura 5.32: Prueba sobre polígono de forma compleja 5 (Martín López, 2015b).

## 5.2. Método basado en Técnicas Metaheurísticas. Primera Aproximación

En esta sección se desarrolla la primera aproximación al Algoritmo Genético propuesto, planteando un estudio teórico previo a su implementación.

En la Sección 5.1, se ha hallado un método capaz de obtener un número razonable de áreas alrededor de un edificio que permitan la vigilancia completa del mismo. Como se ha mencionado en las conclusiones obtenidas (Apartado 5.1.6), este método puede ser útil bajo ciertas circunstancias: los vigilantes no tienen un alcance limitado y el ángulo de la visual con respecto al paramento no es relevante.

No obstante, en la práctica real de los trabajos de toma de datos geométricos para la realización de un levantamiento arquitectónico, no es común que este trabajo pueda realizarse sin atender a esas dos restricciones. Los instrumentos que se utilizan (estaciones totales de topografía, cámaras fotográficas o escáneres láser), tienen un rango de trabajo determinado, comprendido entre una distancia mínima y una distancia máxima. Por otro lado, la calidad de los datos obtenidos está estrechamente relacionada con el ángulo que forma la visual con el paramento leído.

Por lo tanto, tras haber desarrollado el algoritmo anterior, se ha creído necesario afinar las posiciones de estos puntos de vista, atendiendo a las restricciones anteriormente mencionadas, además de algunas otras más.

Llegados a este punto, puede apreciarse que el campo de búsqueda de la solución óptima se amplía, y al introducir nuevas restricciones se hace impracticable diseñar una forma manual de hallar una solución lo suficientemente buena.

Tras exponer el problema a expertos en Geometría Computacional, se sugirió aplicar alguna técnica de optimización metaheurística, en concreto la conocida como **Algoritmos Genéticos**, AGs. Esta técnica ya se ha descrito ampliamente en la Sección 4.2, y genera como resultado una solución lo suficientemente buena como para ser considerada óptima. Recordando la naturaleza del problema, el cual es *NP-duro*, como se ha mencionado en la Sección 1.4, hallar un método que obtenga una solución suficientemente buena es lo máximo que puede lograrse.

Para llevar a cabo el diseño del Algoritmo Genético, se ha pasado por distintas fases. En esta sección se desarrolla una primera aproximación del diseño

del AG, en la cual se identifican los elementos que van a formar parte de él. Los resultados de este apartado han sido presentados en el IX Encuentro Andaluz de Matemática Discreta, celebrado en Almería, en Octubre de 2015 (Cabrera Revuelta et al., 2015b).

Dada la planta de un edificio, y una serie de puntos a su alrededor, una solución al problema de vigilar el contorno del edificio desde el exterior del mismo, sería cualquier conjunto de puntos que cubran la vigilancia de tal edificio (Véase Figura 5.33). Hablando en términos de AGs, ese conjunto de posicionamientos es considerado un **Individuo**.

Los distintos puntos situados alrededor del edificio que aparecen reflejados en la Figura 5.33 es, en sí, una solución al problema que se plantea, pero, obviamente, no es una solución óptima, pues muchos de estos puntos son innecesarios, ya que proporcionarían información redundante.

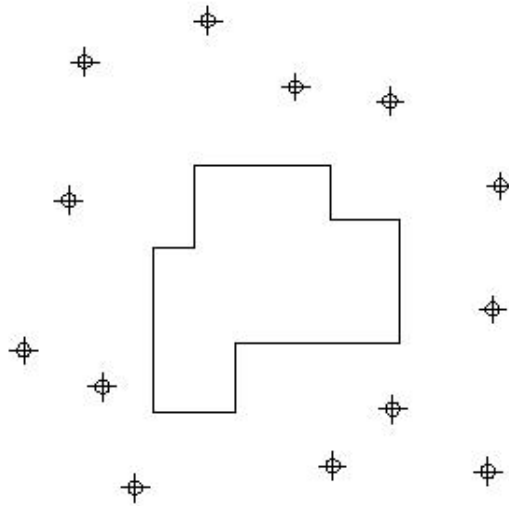


Figura 5.33: Posible solución, o individuo. Elaboración Propia.

Cabe destacar que es común caer en la equivocación de pensar que un *Individuo* representa a un único posicionamiento. Por naturaleza, no es fácil asociar a un conjunto de posicionamientos con el término *Individuo*, pues se tiende a pensar en un individuo como una entidad única, una unidad. Por lo que el lector no debe perder de vista, de aquí en adelante, que el término *Individuo* hace referencia a un conjunto de posicionamientos, y no a un único posicionamiento.

En esta primera aproximación se piensa que, conocidas las áreas de visibilidad obtenidas a través del algoritmo desarrollado en el Apartado 5.1.2, sería provechoso partir de esa información para crear individuos, favoreciendo a aquellos que se ubiquen en las áreas encontradas.

### 5.2.1. Elementos de los Algoritmos Genéticos

A continuación, se describe cada uno de los elementos que forman parte de los llamados Algoritmos Genéticos aplicados al problema planteado.

- **Individuo**,  $I_i \rightarrow$  Un individuo es una posible solución al problema de vigilar un edificio.

Un individuo está formado por un número de **Componentes**. Cada componente representa un posicionamiento o punto de vista.

- **Componentes de un individuo**  $\rightarrow$  Son los puntos alrededor del edificio desde los cuales se realiza la toma de datos, es decir, son los puntos sobre los que se colocan los instrumentos para realizar las mediciones necesarias. Un individuo  $I_i$  con  $k$  componentes está formado por  $C_1, C_2, \dots, C_{k-1}, C_k$  componentes, o puntos de vista.

El número de componentes de los que consta un individuo está limitado entre 2 y  $\lceil n/2 \rceil$ , siendo  $n$  el número de paramentos a vigilar, o el número de aristas del polígono.

La justificación de estos valores es la siguiente: el polígono más sencillo susceptible de ser vigilado es el triángulo. Un triángulo es correctamente vigilado desde dos puntos de vista, es decir con un individuo de 2 componentes. Por otro lado, se considera que con un punto de vista por cada dos fachadas se cubre la vigilancia del edificio.

$$2 \leq C_k \leq \lceil \frac{n}{2} \rceil$$

- **Gen**  $\rightarrow$  En el problema que se desea optimizar, tiene sentido que los genes sean tales que permitan obtener información acerca de la posición en el espacio de los componentes, y de los paramentos que cada uno de ellos vigila. Al fin y al cabo, la función que se desea optimizar trata de minimizar los puntos de vista y de maximizar la superficie vigilada.
- **Población inicial**  $\rightarrow$  La Población Inicial está compuesta por  $2n$  individuos, siendo  $n$  el número de fachadas a vigilar. Este tamaño de la población puede variar, pero se considera que, por un lado, es un número

suficientemente grande, lo cual crea diversidad en las posibles soluciones de las que partir; además, es proporcional a la complejidad del edificio a vigilar; y, por último, no es un tamaño excesivo que dificulte la correcta ejecución del algoritmo.

- **Probabilidad de Cruce**, o también denominada Tasa de Cruce,  $P_c \rightarrow$  Es la probabilidad que se establece de que al unir dos individuos, el proceso de reproducción resulte satisfactorio. Esta tasa suele tener un valor elevado, de más del 80 %.
- **Probabilidad de Mutación**, o también denominada Tasa de Mutación,  $P_m \rightarrow$  Es la probabilidad que se establece de que al crear un nuevo individuo, éste sufra alguna alteración en su código genético, incorporando en él algún elemento que no esté relacionado con ninguno de sus antecesores. Esta tasa suele ser muy baja en todos los algoritmos genéticos, como máximo, de 10 %. Esto es debido a que el fenómeno de la mutación no es algo que ocurra con frecuencia en la naturaleza.

Una vez identificados los elementos que componen el AG diseñado, se procede a explicar en profundidad cada uno de los procesos que forman parte de él.

### 5.2.2. Creación de la Población Inicial

Al iniciar la población se establece una forma de crearla, es decir, una forma de elegir quiénes van a ser los individuos que van a conformar la Población Inicial.

El tamaño de la población de un AG debe cumplir con dos condiciones. Por un lado, no puede ser tan pequeño que no ofrezca variedad para cubrir el espacio de búsqueda. Pero por otro lado, no puede ser tan grande que encarezca el coste computacional de la operación, esto es, el tiempo de ejecución del algoritmo.

Por ello se establece que el tamaño es de  $2n$  individuos, como se ha comentado anteriormente, siendo  $n$  el número de paramentos a vigilar.

En esta primera aproximación, la idea es aplicar este algoritmo tras haber obtenido las áreas de visibilidad siguiendo el método expuesto en el Apartado 5.1.2.



Habiéndose determinado  $k$  áreas de visibilidad, pueden crearse individuos con  $k$  componentes, de forma que a cada área de visibilidad le corresponda un componente. De este modo se garantiza que el edificio queda vigilado por completo. Al partir de las áreas de visibilidad obtenidas tras la aplicación del Algoritmo 5.1.2, es posible crear individuos que presenten un buen comportamiento ante el problema. Sin embargo, el número de individuos situados sobre estas áreas de visibilidad debe estar limitado. Dar *demasiadas pistas* al algoritmo genético puede conducir a una rápida convergencia del mismo; y lo más importante, a no explorar el campo de búsqueda de una forma exhaustiva.

Es por ello que en la población inicial habrá, como máximo,  $\lceil 10(\frac{2n}{100}) \rceil$  individuos que tendrán un componente en cada una de las áreas de visibilidad obtenidas con anterioridad, es decir, un 10 % de los individuos de la población inicial.

Al iniciar el algoritmo, se preguntará al mismo si el primer individuo es, o no, un individuo con sus componentes ubicados en las áreas de visibilidad obtenidas tras aplicar el Algoritmo 5.1.2. La probabilidad de que sí lo sea será de un 80 % frente a un 20 % de posibilidades de que no lo sea.

Se repite este proceso  $\lceil 10(\frac{2n}{100}) \rceil - 1$  veces.

El 80 % de los individuos restantes, tendrán sus componentes ubicados en cualquier punto del exterior del polígono.

Para los individuos que no tengan sus componentes ubicados sobre las áreas de visibilidad previamente encontradas, se establece aleatoriamente el número de componentes del individuo, así como la ubicación de los mismos. Como se ha mencionado anteriormente, el número de componentes que posee un individuo oscila entre 2 y  $\lceil n/2 \rceil$ , siendo  $n$  el número de vértices de los que consta el polígono que representa el recinto exterior del edificio en planta.

Al finalizar se habrá obtenido una población de  $2n$  individuos.

### 5.2.3. Codificación

Una correcta **Codificación** permite identificar y diferenciar a un individuo con respecto a otro. Se ha decidido que la forma más útil de codificar a un individuo es mediante las coordenadas polares de sus componentes. Tomando como origen del sistema de referencia el centro del polígono que representa la planta del edificio objeto de estudio, resulta muy fácil ubicar a los componentes a su alrededor, así como establecer relaciones entre los componentes.

De esta forma, un individuo  $I_i$  compuesto por  $C_k$  componentes, queda expresado de la siguiente forma:

$$I_i \rightarrow (Posic._1; Posic._2; \dots; Posic._{k-1}; Posic._k)$$

$$I_i \rightarrow (C_1; C_2; \dots; C_{k-1}; C_k)$$

$$I_i \rightarrow (\alpha_{i1}, r_{i1}; \alpha_{i2}, r_{i2}; \dots; \alpha_{i(k-1)}, r_{i(k-1)}; \alpha_{ik}, r_{ik})$$

Siendo  $\alpha_{ik}$  y  $r_{ik}$  las coordenadas polares del Componente  $k$  perteneciente al Individuo  $i$ .

#### 5.2.4. Función Objetivo y Valor de Ajuste

Como ya se ha mencionado en el Apartado 4.2.5.1 el aspecto más importante a estudiar en la confección de un Algoritmo Genético, es el diseño de la **Función Objetivo**, o también denominada *Función Fitness*.

El resultado de aplicar esta función a un individuo es un valor, denominado **Valor de Ajuste**, el cual se describe en el Apartado 4.2.5.2. En esta primera aproximación, se pretende que el valor sea lo más cercano a 0.

Las variables que forman parte de la Función Objetivo son las siguientes:

- **Número de fachadas vigiladas,  $n_v$ .**

Una solución óptima es aquella que vigila a todas las fachadas de un edificio, de forma que el valor de  $n_v$  fuera igual al valor de  $n$ , o dicho de otra forma,  $n_v = n$

Para que este parámetro tienda a cero al vigilar todas las fachadas, se expresa de la siguiente forma:

$$1 - \frac{n_v}{n} \tag{5.1}$$

- **Número de componentes,  $k$ .**

El número de componentes que conforma un individuo está comprendido entre 2 y  $\lceil n/2 \rceil$ . Una solución óptima es aquella que tenga un número de componentes reducido, siendo el mínimo 2. Cabe resaltar que sólo en el caso de que la planta del edificio fuera de forma triangular, bastaría con dos componentes para su completa vigilancia. Por lo tanto, no se puede penalizar de forma lineal el incremento de componentes.

Por lo que la forma en que se expresa este parámetro para que tienda a cero, es la siguiente:

$$2^{k-2} - 1 \quad (5.2)$$

■ **Número de componentes conectados,  $K_c$ .**

En el caso de utilizar estaciones totales, o escáneres láser, una vez realizada la toma de datos, es necesario unir los datos obtenidos desde puntos de vista distintos. Estos datos obtenidos son nubes de puntos, más o menos densas, pero, en cualquier caso, son documentos difíciles de manejar. Este proceso de unión de nubes de puntos, o registro, se realiza en la Fase de Gabinete.

Con el fin de garantizar que la unión entre las distintas nubes de puntos se haga de forma correcta, se establece como parámetro favorable que exista visibilidad directa entre los distintos posicionamientos (componentes). Esto quiere decir, que dentro del campo de visión de un componente, exista otro componente.

No obstante, no se trata de un requisito fundamental, ya que la unión de los datos es también posible en el caso de que dos posicionamientos tengan una zona de visibilidad común lo suficientemente amplia.

Existiendo  $k$  componentes, se considera que para unir esos componentes se necesitan  $k$  aristas. El valor de componentes conectados  $k_c$  será el número de uniones que sí es posible trazar entre los componentes del individuo sin que atravesase ningún obstáculo, o el interior del propio edificio (Véase Figura 5.34). Puede darse el caso de que los componentes puedan unirse entre sí de distintas formas, no teniendo porqué tratarse de dos componentes consecutivas. En cualquier caso el valor  $k_c$  no puede exceder al valor de  $k$ . Este parámetro se expresará de la siguiente forma, para que tienda a cero:

$$1 - \frac{k_c}{k} \quad (5.3)$$

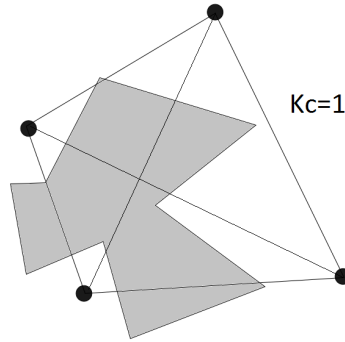


Figura 5.34: En el ejemplo expuesto, existen cuatro componentes. Se observa que de las rectas trazadas, sólo una no atraviesa el edificio. El individuo constituido por cuatro componentes tiene por valor  $k_c = 1$ . Elaboración Propia.

■ **Valor del Ángulo de Incidencia,  $V\beta$ .**

Se refiere al ángulo con el que un componente vigila a una fachada, es decir, el ángulo formado entre el paramento a vigilar y el rayo de visión emitido desde el instrumento que realiza la toma de datos hasta el punto leído. Este dato revela la calidad con la que se realiza la toma de datos. Para entender este concepto, la vigilancia de un paramento que se encuentra totalmente perpendicular a la vista del vigilante, no es adecuada. En ese caso, el ángulo formado entre la visual y el paramento es de  $0^\circ$ .

Para que la solución sea óptima, todas las fachadas deben ser vigiladas con un ángulo mayor o igual a un **Ángulo Mínimo Admisible**, el cual varía en función al instrumento con el que se realice el levantamiento (estación topográfica, cámara de fotos o escáner láser) y también en función a las características propias del instrumento a utilizar. Por lo tanto, si el ángulo de incidencia es mayor o igual al valor del ángulo mínimo admisible, se establece que la fachada está siendo correctamente vigilada.

Por lo que este parámetro se traduce como una n-upla de  $n$  valores para un individuo  $i$ , de la siguiente forma:

$$(V\beta_{i1}, V\beta_{i2}, \dots, V\beta_{i(n-1)}, V\beta_{in}) \quad (5.4)$$

Estos valores serán 0 o 1, según el siguiente criterio:

- $\beta_{in_j} \rightarrow$  Ángulo mayor con el que un individuo  $i$  vigila a una fachada  $j$  en su punto más desfavorable. Es decir, el ángulo desde el componente al punto más lejano de dicha fachada.
- $\gamma_{adm} \rightarrow$  Ángulo a partir del cual se considera que el ángulo de vigilancia es admisible.

$$\gamma_{adm} - \beta_{in_j} = \begin{cases} \leq 0 \rightarrow V\beta_{in_j} = 0 \\ > 0 \rightarrow V\beta_{in_j} = 1 \end{cases} \quad (5.5)$$

En la Función Objetivo este parámetro se introduce como la suma de los valores de todas las fachadas, siendo, de esta forma, el valor óptimo el más cercano a 0.

$$\sum_{i=1}^n V\beta_{in} = V\beta_{i1} + V\beta_{i2} + \dots + V\beta_{in} \quad (5.6)$$

Llegados a este punto, cabe destacar que no todas estas variables son igual de importantes, al igual que existen variables que están relacionadas con otras. Por ello, no se le asigna a todas el mismo peso dentro de la Función Objetivo, variando así los valores de  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$  y  $\lambda_4$ .

$$F(i) = \lambda_1 \left(1 - \frac{n_v}{n}\right) + \lambda_2 (2^{k-2} - 1) + \lambda_3 \left(1 - \frac{k_c}{k}\right) + \lambda_4 \sum_{i=1}^n V\beta_{in} \quad (5.7)$$

El número de fachadas vigiladas  $n_v$  y el número de componentes  $k$  son dos parámetros que están estrechamente relacionados, pues si los componentes no están homogéneamente distribuidos, se vigilan menos fachadas. Por ello, se asigna un mayor peso al parámetro  $n_v$  y un peso menor a  $k$ , ya que, aunque es conveniente que haya el menor número de componentes posible, se conoce que el valor 2 no es siempre un valor razonable.

Como se ha comentado anteriormente, la variable que determina el número de componentes conectadas,  $k_c$  no es de importancia fundamental. Esto es debido a que para unir nubes de puntos basta con que dos componentes compartan una zona de visibilidad amplia. Por ello, se le asigna un peso menor a esta parte de la función.

La variable que tiene en cuenta el ángulo con el que se vigila una fachada, sí es una variable de gran trascendencia, pues es la única que evalúa la calidad con la que se está realizando la toma de datos.

Por ello, la función objetivo que se propone es la siguiente:

$$F(i) = 0,4 \left(1 - \frac{n_v}{n}\right) + 0,1 (2^{k-2} - 1) + 0,1 \left(1 - \frac{k_c}{k}\right) + 0,4 \sum_{i=1}^n V \beta_{in} \quad (5.8)$$

#### 5.2.4.1. Comparación entre individuos

Para hacer visible un ejemplo de evaluación de individuos de una población, a continuación se muestra la planta de un edificio de  $n = 11$  vértices y, por tanto, 11 paramentos a vigilar. Se analizan los valores de ajuste de tres posibles individuos que vigilan este edificio.

Se establece que el ángulo mínimo admisible a partir del cual se vigila de forma correcta un edificio,  $\gamma_{adm}$ , es de  $20^\circ$ .

El vértice asignado con el número 1 es el de mayor ordenada, y a igualdad de ordenada, el de menor abscisa. El paramento adyacente a  $v_1$  hacia el sentido horario es el paramento número 1, continuando con la numeración de forma ascendente hacia el sentido de las agujas del reloj, hasta nombrar los  $n$  paramentos, del uno al once.

**Individuo 1** El primer individuo  $I_1$  (Véase Figura 5.35) está compuesto por cuatro componentes.

Los distintos parámetros que conforman la Función Objetivo alcanzan los siguientes valores.

- Número de fachadas vigiladas:  $n_v = 11$ . Todos los paramentos son vigilados
- Número de componentes:  $k = 4$ .
- Número de componentes conectados:  $K_c = 3$ . Existen tres conexiones entre los componentes. Son las líneas representadas de color azul.

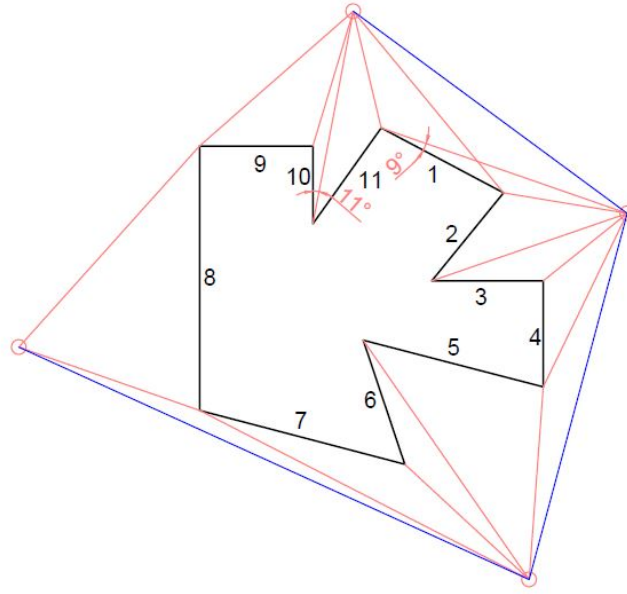


Figura 5.35: Edificio siendo vigilado por el individuo  $I_1$ . Elaboración Propia.

- Valor del Ángulo de Incidencia: De los once paramentos, hay dos que son vigilados con un ángulo menor al ángulo máximo admisible, estos son el paramentos 1 y 10. Pero el paramento 1 es vigilado por otro componente con un mejor ángulo, por lo tanto, los valores serían los siguientes:

$$V\beta_{1-1} = 0, V\beta_{1-2} = 0, \beta_{1-3} = 0, \beta_{1-4} = 0, \beta_{1-5} = 0,$$

$$\beta_{1-6} = 0, \beta_{1-7} = 0, \beta_{1-8} = 0, \beta_{1-9} = 0, \beta_{1-10} = 1, \beta_{1-11} = 0,$$

De esta forma, al ser  $V\beta_1$  la suma de esos valores, su valor final es  $V\beta_1 = 1$ .

Tras aplicar la Función Objetivo 5.8, el valor de ajuste obtenido sería el siguiente:

$$F(1) = 0,4 \left(1 - \frac{11}{11}\right) + 0,1 (2^{4-2} - 1) + 0,1 \left(1 - \frac{3}{4}\right) + 0,4(1) \quad (5.9)$$

$$F(1) = 0 + 0,3 + 0,025 + 0,4 = 0,725 \quad (5.10)$$

**Individuo 2** De la misma forma que se ha realizado con el primer individuo, se procede a analizar cada uno de los parámetros que forman parte de la Función Objetivo para hallar el valor de ajuste del individuo  $I_2$ .

En este caso, el individuo también consta de cuatro componentes, pero la distribución de los mismos no es tan favorable como en el caso anterior (Véase Figura 5.36).

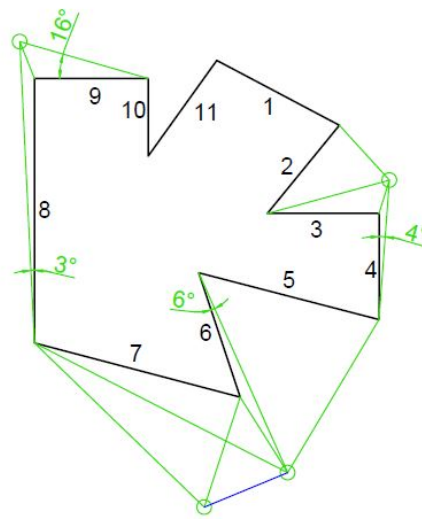


Figura 5.36: Edificio siendo vigilado por el individuo  $I_2$ . Elaboración Propia.

- Número de fachadas vigiladas:  $n_v = 8$ . Los paramentos 1, 10 y 11 no son vigilados.
- Número de componentes:  $k = 4$ .
- Número de componentes conectados:  $K_c = 1$ . Existe tan solo una conexión de las cuatro posibles.
- Valor del Ángulo de Incidencia: De los once paramentos, tan solo ocho están siendo vigilados. Además, cuatro de ellos no se están vigilando con



## 5.2. MÉTODO BASADO EN TÉCNICAS METAHEURÍSTICAS 1ª APROX.

un ángulo mayor al ángulo mínimo admisible. Estos son los paramentos 4, 6, 8 y 9. Estos son los valores:

$$V\beta_{2-1} = 1, V\beta_{2-2} = 0, \beta_{2-3} = 0, \beta_{2-4} = 1, \beta_{2-5} = 0,$$

$$\beta_{2-6} = 1, \beta_{2-7} = 0, \beta_{2-8} = 1, \beta_{2-9} = 1, \beta_{2-10} = 1, \beta_{2-11} = 1,$$

De esta forma, al ser  $V\beta_2$  la suma de esos valores, su valor final es  $V\beta_2 = 7$ .

$$F(2) = 0,4 \left(1 - \frac{8}{11}\right) + 0,1(2^{4-2} - 1) + 0,1 \left(1 - \frac{1}{4}\right) + 0,4(7) \quad (5.11)$$

$$F(2) = 0,112 + 0,3 + 0,075 + 2,8 = 3,287 \quad (5.12)$$

**Individuo 3** Por último, se procede a analizar un tercer individuo,  $I_3$ , representado en la Figura 5.37.

- Número de fachadas vigiladas:  $n_v = 9$ . Los paramentos 2 y 3 no son vigilados.
- Número de componentes:  $k = 6$ .
- Número de componentes conectados:  $K_c = 6$ . En la Figura 5.37 aparecen las conexiones de color rojo.
- Valor del Ángulo de Incidencia: De los once paramentos, están siendo vigilados nueve. De estos nueve, los paramentos 4, 8 y 10 no están siendo vigilados con el ángulo adecuado.

$$V\beta_{3-1} = 0, V\beta_{3-2} = 1, \beta_{3-3} = 1, \beta_{3-4} = 1, \beta_{3-5} = 0,$$

$$\beta_{3-6} = 0, \beta_{3-7} = 0, \beta_{3-8} = 1, \beta_{3-9} = 0, \beta_{3-10} = 1, \beta_{3-11} = 0,$$

De esta forma, al ser  $V\beta_3$  la suma de esos valores, su valor final es  $V\beta_3 = 5$ .

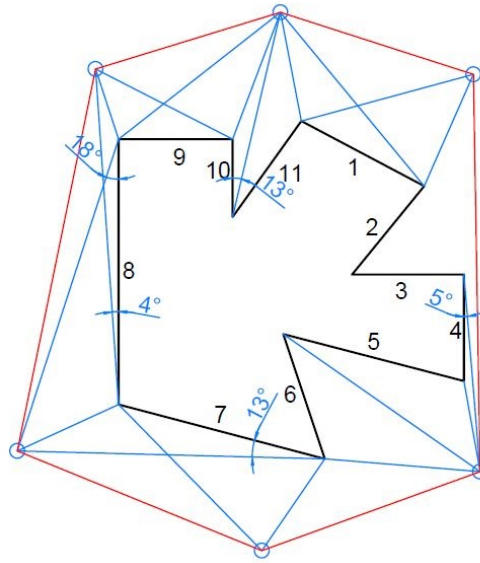


Figura 5.37: Edificio siendo vigilado por el individuo  $I_3$ . Elaboración Propia.

Tras aplicar la Función Objetivo 5.8, el valor de ajuste obtenido es el siguiente:

$$F(3) = 0,4 \left( 1 - \frac{9}{11} \right) + 0,1 (2^{6-2} - 1) + 0,1 \left( 1 - \frac{6}{6} \right) + 0,4(5) \quad (5.13)$$

$$F(3) = 0,07 + 1,5 + 0 + 2 = 3,57 \quad (5.14)$$

De los tres individuos estudiados, el que mejor se adapta al problema es el primero de ellos  $I_1$ , pues visualiza a todos los paramentos, la conexión entre los componentes es adecuada, y la mayoría de los paramentos son vigilados con un buen ángulo de visión.

### 5.2.5. Selección

Llegados a este punto, el siguiente paso es seleccionar a los individuos más aptos, como se ha explicado en el Apartado 4.2.6.1. Estos individuos seleccionados crearán la descendencia más adelante. Además, los mejores individuos

## 5.2. MÉTODO BASADO EN TÉCNICAS METAHEURÍSTICAS 1ª APROX.

de entre los seleccionados serán individuos *longevos*, ya que pasarán automáticamente a la siguiente generación.

De la Población Inicial se selecciona a un 80 % de sus individuos. Además, de entre estos individuos seleccionados, el 20 % mejor tendrá posibilidad de pasar a la siguiente generación, como se muestra en la Figura 5.38.



Figura 5.38: Subconjuntos de individuos de la población inicial tras el proceso de Selección. Elaboración Propia.

El proceso de **Selección** se lleva a cabo por **Torneo**, explicado en el Apartado 4.2.6.1. De esta forma, el proceso de selección consiste en seleccionar de forma aleatoria a dos individuos, y de entre ellos, se escogerá al que mejor valor de ajuste posea. El individuo que ha sido seleccionado pasa a un subconjunto de la población, y el individuo no seleccionado vuelve con el resto de individuos, pudiendo ser de nuevo escogido de forma aleatoria para volver a competir con otro individuo.

Este proceso es iterativo, y se repite hasta que un 80 % de los individuos de la Población Inicial haya sido seleccionado. Estos individuos seleccionados serán ordenados por su valor de ajuste, y de entre ellos (teniendo en cuenta que el mejor valor de ajuste es el cero), el 20 % con el mejor valor de ajuste es además susceptible de sobrevivir a la siguiente generación.

### 5.2.6. Cruce

Tras haber seleccionado los individuos mediante la Selección por Torneo, es momento de crear la segunda generación a partir del proceso de reproducción o cruce entre los individuos seleccionados. Este proceso ha sido explicado en el Apartado 4.2.6.2.

Se seleccionan aleatoriamente de entre los individuos seleccionados a dos individuos. Posteriormente, se aplica la **Probabilidad de Cruce**  $P_c$ , que deter-

mina si existe o no reproducción. Esta probabilidad es de un 80 %.

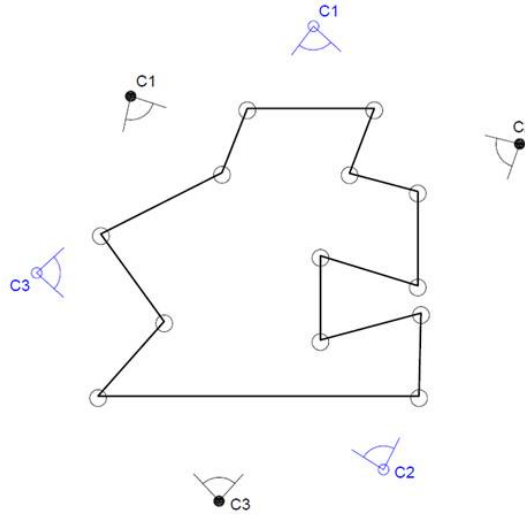


Figura 5.39: Dos individuos  $I_1$  e  $I_2$  compuestos por sus respectivos componentes, en azul y en negro. Elaboración Propia.

En el caso de que sí proceda la reproducción, se genera aleatoriamente el número de componentes de los que consta el nuevo individuo. Se divide el espacio en  $k$  sectores de igual tamaño, tomando como centro el origen del sistema de coordenadas polares, es decir, el centro del polígono (Véase Figura 5.40). El comienzo de la división es en algún valor entre 0 y  $\frac{\pi}{4}$ .

En cada sector se ubica a un componente que, o bien, es combinación lineal de los componentes de los padres, o bien, si no hay ningún componente de alguno de los padres en ese sector, se escoge de forma aleatoria (Véase Figura 5.41).

Dado que el tamaño inicial de la población es de  $2n$ , se repite el proceso de cruce hasta obtener  $2n$  nuevos individuos.

### 5.2.7. Mutación

Como se describe en el Apartado 4.2.6.3, la mutación de un individuo provoca que alguno de sus genes, generalmente uno sólo, varíe su valor de forma aleatoria.

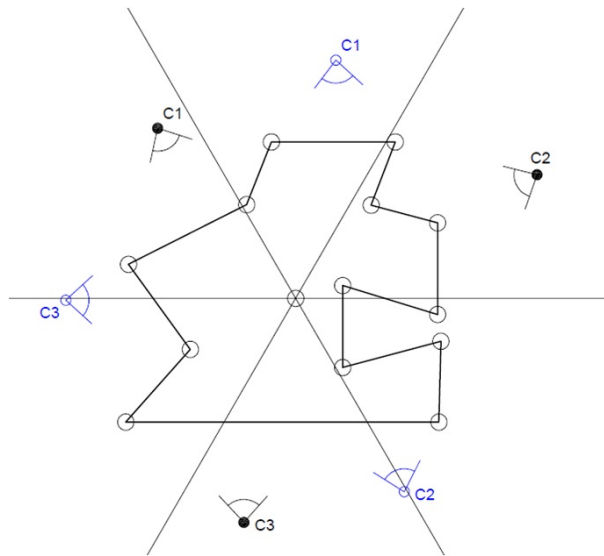


Figura 5.40: División del espacio en  $k = 6$  sectores. El nuevo individuo está compuesto por seis posicionamientos. Elaboración Propia.

La Tasa de Mutación o Probabilidad de Mutación,  $P_m$ , se establece en 0,08, ya que interesa que la población sea diversa. Se aplica esta tasa a cada uno de los nuevos individuos. En el caso de que procediese la mutación, se elige de forma aleatoria a un componente del individuo y se cambia su posición de forma también aleatoria, o se elimina.

### 5.2.8. Creación de la Siguiente Generación

En este momento se cuenta con  $2n$  nuevos individuos, que ya han pasado por el proceso de mutación, si se ha dado el caso. Además, también se había reservado el 20 % de los individuos de la población anterior seleccionados, con el mejor valor de ajuste, y los cuales pueden ser copiados como se ha descrito en el Apartado 4.2.6.4. Por lo tanto, existen  $2n + 0,32n = 2,32n$  individuos que pueden formar la segunda generación, siendo  $n$  el número de parámetros a vigilar.

Estos  $2,32n$  individuos se ordenan por su valor de ajuste, y los  $2n$  individuos mejores son los que conforman la segunda generación.

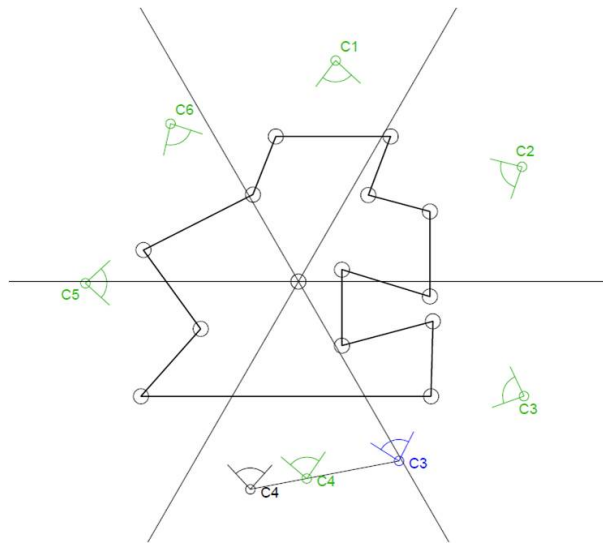


Figura 5.41: Los componentes del nuevo individuo se crean de distintas formas:  $C_1$ ,  $C_2$ ,  $C_5$  y  $C_6$  se ubican en la misma posición donde estuvo un componente de alguno de sus antecesores.  $C_3$  se crea de forma aleatoria.  $C_4$  es combinación lineal de sus antecesores. Elaboración Propia.

### 5.2.9. Creación de Generaciones Sucesivas y Fin

Una vez que se ha pasado de la Población Inicial a la Segunda Generación, se debe repetir el proceso para crear tantas generaciones como sean necesarias hasta que el algoritmo pare o, dicho de otro modo, converja, atendiendo a lo descrito en el Apartado 4.2.7.

### 5.2.10. Pseudocódigo

A continuación se ha redactado el pseudocódigo del estudio teórico previo para el Algoritmo Genético que se ha planteado en la presente sección.

#### 1. Población Inicial.

a) Primer individuo. Individuo cuyos componentes pertenecen a las Áreas de Visibilidad. Sí  $\rightarrow 0,8$  No  $\rightarrow 0,2$

1) Sí  $\rightarrow$  Se elige un punto situado en cada una de las áreas de forma aleatoria, para situar en cada una de ellas a un componente.

- 2) No  $\rightarrow$ 
  - $a'$  Establecer el número de componentes, comprendido entre 2 y  $\lceil n/2 \rceil$ , de forma aleatoria.
  - $b'$  Se eligen tantos componentes como haya resultado del Paso 1a2a, de forma aleatoria.
- 3) Se repite el Paso 1a  $\lceil 10 \left( \frac{2n}{100} \right) \rceil - 1$  veces.
- 4) Se repite el Paso 1a2 tantas veces como sea necesario hasta completar la población de  $2n$ , es decir, se repite 0,  $9(2n)$  veces, o expresado de otra forma  $2n - \lceil 10 \left( \frac{2n}{100} \right) \rceil$  veces.

2. Selección.

- a) Aplicar la Función Objetivo a los individuos de la Población Inicial, y obtener así el Valor de Ajuste de cada individuo.
- b) Selección por Torneo.
  - 1) Elegir aleatoriamente a dos individuos de la Población Inicial.
  - 2) El individuo que mejor Valor de Ajuste posea, pasa a la lista de *Individuos Seleccionados*. El otro individuo vuelve con el resto de la población no seleccionada.
- c) Repetir el Paso 2b hasta haber seleccionado 1,  $6n$  individuos.
- d) Todos estos individuos pasan a formar parte del conjunto *Individuos Seleccionados*.
- e) Ordenar los 1,  $6n$  individuos seleccionados, por su valor de ajuste.
- f) Los 0,  $32n$  individuos que presenten un valor de ajuste mejor (más cercano a cero) pasan a formar parte de un subconjunto denominado *Posibles Individuos Longevos*.

3. Cruce.

- a) Creación del primer par de individuos.
  - 1) Selección aleatoria de dos individuos, entre los que fueron seleccionados en el Paso 2d.
  - 2) Aplicación de la Probabilidad de Cruce,  $P_c = 0,8$ .
    - $a'$  No  $\rightarrow$  No hay descendencia. Se vuelve al Paso 3a1.
    - $b'$  Sí  $\rightarrow$  Se crean dos individuos. El individuo elegido primero es la madre, y el individuo elegido segundo es el padre.
      - Hijo 1
        - Se establece aleatoriamente el número de componentes del individuo  $k$ , al igual que se hizo en el Paso 1a2a.

- Se divide el espacio en  $k$  sectores, tomando como comienzo de la división un valor entre 0 y  $\frac{\pi}{4}$ .
- Creación de los nuevos componentes por sectores: en cada sector se ubica a un componente. Pueden darse diferentes casos:
  - En el sector no existe ningún componente de madre o de padre: Se elige un punto aleatorio en el sector.
  - En el sector existen componentes de la madre y del padre: Se establece aleatoriamente un punto que sea combinación lineal de ellos.
  - En el sector sólo existe un componente del padre o un componente de la madre: El nuevo componente es el mismo que el del antecesor.

■ Repetir el Paso 3a2b para crear el Hijo 2.

b) Repetir el Paso 3a1 hasta crear el  $2n$  individuos nuevos.

#### 4. Mutación.

- a) Aplicar sobre el primer individuo de los creados en el Paso 3, la  $P_m$  de 0,08.
- 1) Sí procede la mutación → Se escoge aleatoriamente a uno de los componentes y se cambia por otro componente aleatoriamente, cabiendo también la posibilidad de su eliminación.
  - 2) No procede la mutación → El individuo permanece igual.
- b) Repetir el Paso 4a tantas veces como individuos nuevos se hayan creado en el Paso 3.

#### 5. Creación de la siguiente generación.

- a) Ordenar por su valor de ajuste de mejor a mayor los individuos obtenidos en el Paso 2f y los obtenidos tras el proceso de mutación del Paso 4.
- b) Los  $2n$  individuos con menor valor de ajuste son los que componen la nueva generación.

#### 6. Iteraciones.

- a) Creación de una Generación.
- 1) Repetición del Paso 2 *Selección* con los individuos de la última generación definidos en el Paso 5.



## 5.2. MÉTODO BASADO EN TÉCNICAS METAHEURÍSTICAS 1ª APROX.

---

- 2) Repetición del Paso 3 *Cruce* con los individuos seleccionados en el Paso 6a1.
- 3) Repetición del Paso 4 *Mutación* con los individuos resultantes del Paso 6a2.
- 4) Definición de la *Nueva Generación* según el Paso 5
- b) Iteración del Paso 6a, sustituyendo en el Paso 6a1 los individuos del Paso 5 por los de la *Nueva Generación* resultantes del Paso 6a4.
- c) Parar las iteraciones cuando después de  $100n$  la media del *Valor de Ajuste* de la Generación no haya disminuido, o bien, en el momento en que se hayan realizado  $1000n$  iteraciones.

7. Fin

### 5.3. Método basado en Técnicas Metaheurísticas. Segunda aproximación e implementación

Tras haber realizado una primera aproximación del diseño de un Algoritmo Genético para minimizar los posicionamientos desde los cuales vigilar el contorno de un edificio (Sección 5.2), es necesario pasar al siguiente paso: implementación y comprobación de resultados.

En esta sección, el algoritmo es repensado: se parte del expuesto anteriormente, pero se introducen algunas modificaciones y restricciones, como el alcance limitado de los vigilantes (Canales Cano, 2004).

La nueva versión del Algoritmo Genético está condicionado por su implementación como un Sistema Multiagente en el lenguaje *Net-Logo*.

En esta sección, en primer lugar se explica en qué consiste el Modelado basado en Agentes (Apartado 5.3.1); posteriormente se hace una breve introducción acerca del funcionamiento del lenguaje de programación *Net-Logo* (Apartado 5.3.2); a continuación, se detalla el diseño del modelo que se ha realizado (Apartado 5.3.3) así como su funcionamiento (Apartado 5.3.4); y, por último, se describen los resultados obtenidos (Apartado 5.3.5), y las principales conclusiones (Apartado 5.3.6).

La idea de desarrollar el AG como un Sistema Multiagente en la plataforma *Net-Logo* surge de la participación en las Jornadas FORMA 15 coordinadas por los profesores Fernando Sancho Caparrini y Joaquín Borrego, de la Universidad de Sevilla. El trabajo realizado en ella se puede encontrar en una publicación de las propias jornadas, (Cabrera Revuelta et al., 2015a).

En dichas jornadas se creó un equipo multidisciplinar con ingenieros informáticos que ha continuado trabajando en el perfeccionamiento del modelo que en esta sección se presenta.

#### 5.3.1. Modelado basado en Agentes

En computación, el concepto de **agente** es referido a entidades autónomas, las cuales interactúan entre ellas, y con su entorno, o medio ambiente en el que viven. Las acciones de los agentes son de corto alcance, pero la interacción de muchos agentes en un entorno da lugar a un fenómeno global. Esas propiedades globales que emergen de la interactuación de los agentes con su entorno, son conocidas como **Propiedades Emergentes**. Ellas constituyen el principal

objetivo del estudio de los Sistemas Multiagente.

Un agente inteligente es una entidad que percibe y actúa sobre un entorno de forma razonada. Es capaz de ejecutar tareas específicas de forma autónoma y de interactuar con su entorno de forma útil (Brenner et al., 1998).

Los **Sistemas Multiagente-SMA**, son sociedades de agentes que se orientan a fines comunes y mediante la coordinación y colaboración, son capaces de distribuir tareas (Mendez et al., 2007). Los agentes, además de solucionar problemas de forma individual, son capaces de, mediante la interacción, resolver problemas que van más allá de sus capacidades individuales o de los conocimientos de cada uno de los resolvedores individuales. Para que la comunicación entre agentes sea posible, se deben sincronizar acciones, enviar y recibir conocimiento, establecer un lenguaje común, así como otros protocolos de actuación.

El **Modelado Basado en Agentes** - ABM sirve para simular el comportamiento de agentes, y solucionar problemas en sistemas complejos a través de técnicas metaheurísticas. Este tipo de modelado se caracteriza por lo siguiente:

- El agente tiene actividad autónoma y lleva a cabo actividades por iniciativa propia.
- Los agentes están dotados de sociabilidad, siendo capaces de interactuar con otros agentes. Un agente no es un ente aislado, sino un elemento de una sociedad.
- El modelado multiagente tiene una alta capacidad de integración y flexibilidad para incorporar nuevos agentes.

#### 5.3.2. Acerca de *Net-Logo*

*Net-Logo* es un entorno programable de modelado para simular fenómenos naturales y sociales. Es especialmente adecuado para modelar sistemas complejos, mediante Sistemas Multiagente, y que se desarrollan en el tiempo (Wilensky, 1999). Este lenguaje tiene una curva de aprendizaje muy rápida, siendo así fácil iniciarse en su programación.

### 5.3.2.1. Tipos de Agentes en *Net-Logo*

Como se ha mencionado anteriormente, los agentes son entidades autónomas, las cuales interactúan entre sí y con el entorno en el que viven.

*Net-Logo* trabaja con tres tipos de agentes. Estos agentes son: el agente móvil, o también conocido como **tortuga**; el medio ambiente, de forma que los *patches* son las porciones de terreno en las que se divide el medio ambiente; y los posibles enlaces que puedan darse entre los agentes móviles, denominados *links*. También existe un cuarto agente: se trata del usuario que maneja el programa a través del código y ejecuta las órdenes.

El **Mundo** en *Net-Logo* que se ha utilizado para modelar el problema es bi-dimensional, y se divide en una cuadrícula o *patches*. Cada *patch* es una porción de forma cuadrada del terreno, y las tortugas se mueven sobre ellos.

### 5.3.3. Diseño del Algoritmo Genético en *Net-Logo*

Como se comentó anteriormente en la Sección 5.2, un Algoritmo Genético parte de un conjunto de posibles soluciones al problema. Esas soluciones son llamadas **Individuos**. En el caso de estudio, un individuo es un conjunto de posicionamientos, desde los cuales, teniendo en cuenta todos los posicionamientos de forma conjunta, es posible vigilar la parte externa de un edificio.

Por lo tanto, en el AG diseñado, un individuo está compuesto por diferentes posicionamientos, los cuales reciben el nombre de **Componentes**.

En esta segunda versión del AG, uno de los cambios que se introduce es que todos los individuos de la población tienen el mismo número de componentes, pero algunos de ellos estarán *activados* y otros no, de forma aleatoria.

Otro cambio a destacar es la inclusión del concepto de **Visibilidad de Alcance Limitado** (Ntafos, 1992), tratado en el Apartado 4.1.2.4. Por lo que se introduce un parámetro en la nueva Función Objetivo que refleja la limitación de alcance del instrumento a utilizar.

A continuación se desarrolla el diseño del algoritmo de forma detallada.

### 5.3.3.1. Definición del Entorno

Con el fin de implementar el AG de forma satisfactoria, es necesario desarrollar el protocolo conocido como **Protocolo ODD** -*Overview, Design Concepts and Details* (Grimm et al., 2010), cuya función es la de ordenar y estructurar los problemas y los agentes que participan en el algoritmo.

El protocolo se encuentra dividido en diferentes secciones. En el Apéndice C, puede encontrarse de forma completa. En esta sección se detallan sólo los apartados más importantes.

#### ■ **Objetivo.**

El objetivo final de este modelo basado en agentes (ABM) es el de minimizar los puntos de vista y determinar la posición de éstos, de tal forma que se garantice la visualización completa de los muros de un edificio, desde el exterior del mismo, para poder ejecutar un levantamiento arquitectónico de forma correcta.

#### ■ **Entidades, Estado de las Variables y Escala de los Agentes.**

- Los Agentes tipo **Tortuga** son:
  - Los vértices que conforman la planta del edificio.
  - Los recorredores, los cuales recorren las fachadas (aristas del polígono) lanzando áreas de visión hacia el lado deseado.
  - Los posibles puntos de vista, o estacionamientos, que vigilan el edificio desde el exterior
- Los Agentes tipo **Link** son las fachadas del edificio, que conectan agentes vértices.
- La planta del edificio está conformada por agentes vértices (*tortuga*) y agentes fachada (*link*).
- Los recorredores son agentes capaces de moverse a lo largo del borde del polígono, es decir, a lo largo de las fachadas del edificio. Saltan de un *patch* a otro que compone la línea de fachada. Desde cada uno de esos *patches* lanzan un área de visión hacia el semiplano desde el cual se desea vigilar. El área de visión es un sector circular limitado por: la distancia mínima  $d$ , la cual representa la distancia de trabajo a partir de la cual el instrumento es capaz de leer datos geométricos de forma correcta; la distancia máxima  $D$ , la cual representa la distancia hasta la cual el instrumento es capaz

de leer datos geométricos de forma correcta; y un ángulo de apertura equivalente a  $2\gamma_{maxadm}$ . El valor  $\gamma_{maxadm}$  representa el valor máximo que puede tomar el ángulo formado entre el vector normal en el punto  $P$  y el rayo de visión desde el instrumento hasta dicho punto  $P$ . En el Apéndice B puede encontrarse un estudio acerca del valor máximo que se recomienda para este ángulo. Todo ello puede verse reflejado en la Figura 5.42. Se debe tener en cuenta que este sector circular es simétrico con respecto al vector normal en el punto  $P$ , siendo  $P$  el punto desde donde el área de visibilidad es lanzada por el recorridor. En caso de que al lanzar el área de visibilidad ésta sea interrumpida por un obstáculo que le impida una correcta visualización a través de la misma, la forma del área de visibilidad se verá alterada, como puede verse en la Figura 5.43.

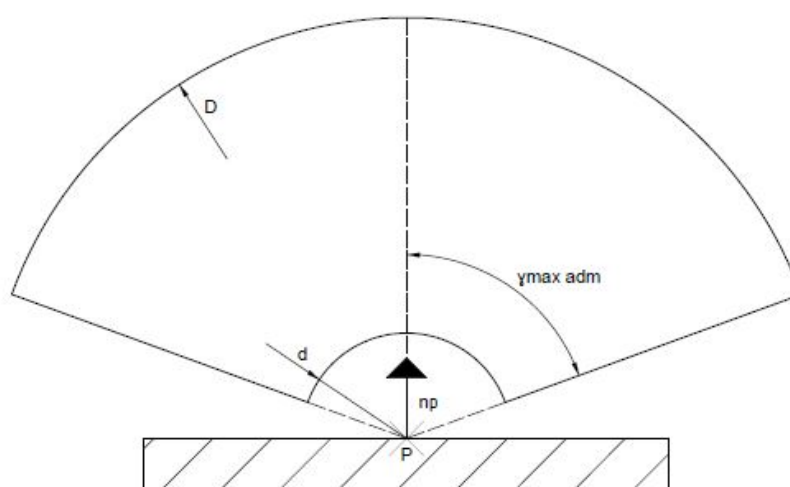


Figura 5.42: Área de visión emitida por cada uno de los recorridores en un *patch* determinado  $P$ . Elaboración Propia.

#### ■ Resumen de Procesos.

- Definición del polígono a vigilar: introducir los vértices en el sentido de las agujas del reloj, definiendo la forma de la planta del edificio.
- Lanzamiento de las áreas de visibilidad: los agentes recorridores se mueven a lo largo de las fachadas, de un vértice a otro, parando en cada *patch* que conforma el borde del edificio. Desde cada posición lanzan un área de visibilidad hacia el exterior, la cual delimita el área desde la cual ese punto de la fachada es vigilado de forma

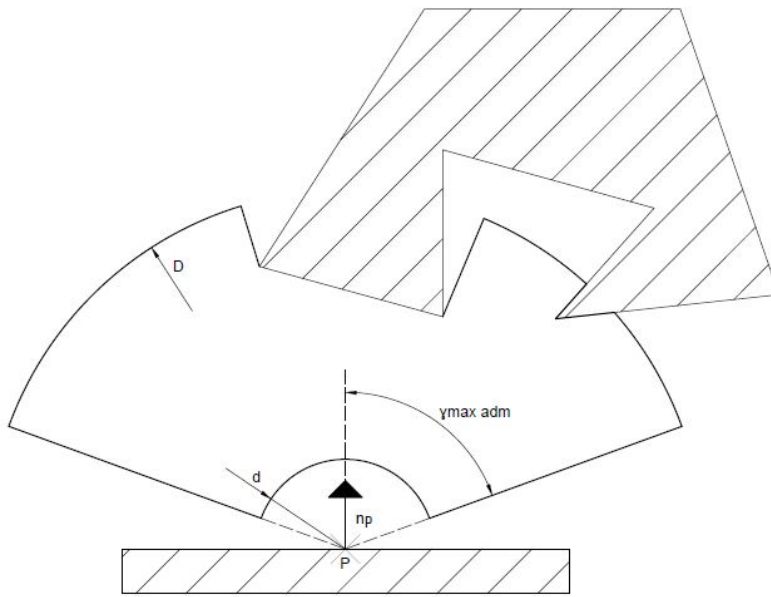


Figura 5.43: Área de visión de un punto P limitada por obstáculos que limitan la visibilidad completa. Elaboración Propia.

correcta. Las dimensiones de las áreas de visibilidad irán en función al tipo de aparato elegido para realizar el levantamiento.

- Evaluación del terreno: consiste en contabilizar la cantidad de áreas de visibilidad que han incidido sobre cada uno de los *patches* del exterior del edificio. Cada *patch* del exterior del edificio es un posible componente de un individuo de la población del AG. Los *patches* del exterior del edificio se colorean en función al número de áreas de visibilidad que pasan sobre él, quedando en color negro los *patches* sobre los que no pasan áreas, y aumentando el color de gris a blanco cuantas más áreas incidan sobre un *patch*. De esta forma, es muy fácil visualizar en la representación del *Mundo* sobre la interfaz de *Net-Logo* a simple vista, cuáles son las zonas sobre las que es más conveniente realizar la toma de datos geométricos.
- Inicialización del Algoritmo Genético: se describe a continuación.
- Parada: se describe de forma detallada en el Apartado 5.3.3.8.

### 5.3.3.2. Población: Codificación

Como ya se ha mencionado, en esta versión del AG todos los individuos que constituyen la población tienen el mismo número de componentes (activados o desactivados) con el fin de hacer más simple computacionalmente su cruce.

El número de componentes de los que consta un individuo es la división del perímetro de la fachada entre la distancia mínima  $d$  a la que trabaja el aparato.

Por otro lado, al tratarse de un mundo bidimensional, una posición es identificada a través de sus coordenadas cartesianas.

Por tanto, la codificación del individuo es de la siguiente forma:

$$I_i \rightarrow (Componente_1; Componente_2; \dots; Componente_{k-1}; Componente_k), \\ (Estado_1; Estado_2; \dots; Estado_{k-1}; Estado_k)$$

$$I_i \rightarrow (X_1, Y_1; X_2, Y_2; \dots; X_{k-1}, Y_{k-1}; X_k, Y_k), \\ (Estado_1; Estado_2; \dots; Estado_{k-1}; Estado_k)$$

El **Estado** puede variar entre 0 (activado), o 1 (desactivado).

De esta forma, todos los individuos están codificados según cadenas de valores del mismo tamaño. Su codificación está compuesta por una primera n-upla de valores de coordenadas cartesianas, y por una segunda n-upla de valores de ceros y unos que se refieren al estado de cada uno de los componentes de la primera n-upla.

En *Net-Logo* el origen de coordenadas corresponde al centro del *patch* central del *Mundo*. Las coordenadas marcan los centros de los *patches*.

### 5.3.3.3. Función Objetivo

La **Función Objetivo** ha variado con respecto a la Función 5.8. El modelo diseñado permite reducir cierta complejidad en el problema. A continuación se estudian los cuatro parámetros que formaban parte de la Función Objetivo inicial, y los cambios que se han producido en esta segunda versión.



En la nueva Función Objetivo no se trata de minimizar a cero el valor de ajuste, sino de maximizar la función. Se premia a aquellos individuos que respondan bien ante el problema, sin que exista un valor de ajuste máximo establecido.

■ **Patches vigilados de la fachada,  $P_v$ .**

Este parámetro sustituye al parámetro  $n_v$  (número de fachadas vigiladas). Esto es debido a que en este modelo es posible estudiar cada *patch* que compone la fachada de forma individual, y no las fachadas como entidades completas. Esto ofrece una gran ventaja, ya que amplía las posibilidades de encontrar una solución óptima.

Este parámetro se contabiliza como la relación entre *patches* vigilados  $P_v$ , y los *patches* totales de la fachada  $P_t$ .

$$\frac{P_v}{P_t} \quad (5.15)$$

■ **Número de componentes activos del individuo,  $K_a$ .**

Esta segunda parte de la Función 5.8 permanece igual, con una pequeña diferencia. En el modelo diseñado todos los individuos tienen el mismo número de componentes,  $K_t$ , pero no todos se encuentran activados. Se trata de cuantificar cuántos componentes están activados, mediante el valor  $K_a$ . Se premia a aquellos individuos cuyo número de componentes activados sea menor, de forma exponencial.

$$2^{K_t - K_a} \quad (5.16)$$

■ **Número de componentes conectados,  $K_c$ .**

La tercera parte de la Función 5.8, permanece igual, salvo que el número de componentes conectados se calcula con respecto a los componentes activos  $K_a$ .

$$\frac{K_c}{K_a} \quad (5.17)$$

■ **Valor del Ángulo de Incidencia,  $V_\beta$ .**

Esta parte de la Función 5.8 desaparece. Al lanzar las áreas de visión desde cada uno de los *patches* que componen la fachada, ya se tiene en cuenta el ángulo a partir del cual un punto es correctamente vigilado.

Por lo tanto, la nueva Función Objetivo está compuesta por tres sumandos, cada uno de los cuales tiene asignado un peso que puede variar.

$$F = P_1 \left( \frac{F_v}{F_t} \right) + P_2 (2^{K_t - K_a}) + P_3 \left( \frac{K_c}{K_a} \right) \quad (5.18)$$

El usuario puede elegir los valores de  $\lambda_1$ ,  $\lambda_2$  y  $\lambda_3$ , siendo recomendable un mayor peso para  $\lambda_1$  y un menor peso para  $\lambda_2$  y  $\lambda_3$ .

En la función del modelo presentado se tienen en cuenta los dos primeros sumandos de la función descrita. Por el momento no se ha introducido el tercer sumando.

#### 5.3.3.4. Selección

La Selección de los individuos de la población del AG se hace por Torneo, como se ha explicado en el Apartado 4.2.6.1, seleccionando aquel individuo que presente un valor de ajuste mayor.

#### 5.3.3.5. Cruce

En el modelo multiagente diseñado, el cruce varía con respecto al que se desarrolla en el Apartado 5.2.6.

Los individuos que han sido seleccionados según su **Valor de Ajuste** son aquellos que pueden generar descendencia. De forma aleatoria, se seleccionan dos individuos de entre los seleccionados, y se aplica la **Probabilidad de Cruce**.

En el caso de que proceda la reproducción, se selecciona un punto de corte de forma aleatoria, y se crean dos nuevos individuos mediante la operación de Cruce en un Punto, explicada en el Apartado 4.2.6.2.

El estado de los componentes, activados o desactivados, varía de forma aleatoria.

#### 5.3.3.6. Mutación

La mutación se aplica a cada uno de los individuos de la población. En el caso de que proceda, la posición de uno de los componentes varía de forma

aleatoria.

#### 5.3.3.7. Creación de la Siguiete Generación

Los mejores individuos de la Población Inicial pueden ser copiados a la siguiente generación. Esos individuos y los generados a partir de las operaciones de cruce y mutación realizadas, se ordenan en función a su Valor de Ajuste.

Aquellos que tengan un mayor Valor de Ajuste forman la siguiente generación, con un tamaño igual al tamaño de la Población Inicial.

#### 5.3.3.8. Convergencia

El proceso de convergencia del algoritmo es el descrito en el Apartado 4.2.7, es decir, después de que el modelo ejecute un número suficientemente alto de iteraciones, aparece la *propiedad emergente del sistema multiagente* (Véase Apartado 5.3.1). En este caso, dicha propiedad emergente representa una solución que puede considerarse como óptima.

En el modelo presentado la convergencia no llega a ejecutarse, pero se visualiza la estabilización de la solución, siendo el usuario quien debe paralizar el AG.

### 5.3.4. Funcionamiento del modelo

A continuación se describe paso a paso el orden en el que deben utilizarse cada uno de los botones y deslizadores del modelo diseñado en *Net-Logo*, haciendo siempre referencia a la interfaz expuesta en la Figura 5.44, la cual representa la parte izquierda de la pantalla del modelo presentado en la documentación digital anexa a esta memoria de tesis.

Paso 1. Introducir el valor  $d$  en la casilla «distancia-minima» → Representa la distancia a partir de la cual el instrumento utilizado puede funcionar correctamente.

Paso 2. Introducir el valor  $D$  en la casilla «distancia-maxima» → Representa la distancia hasta la cual el instrumento utilizado puede realizar la lectura de datos de forma satisfactoria.

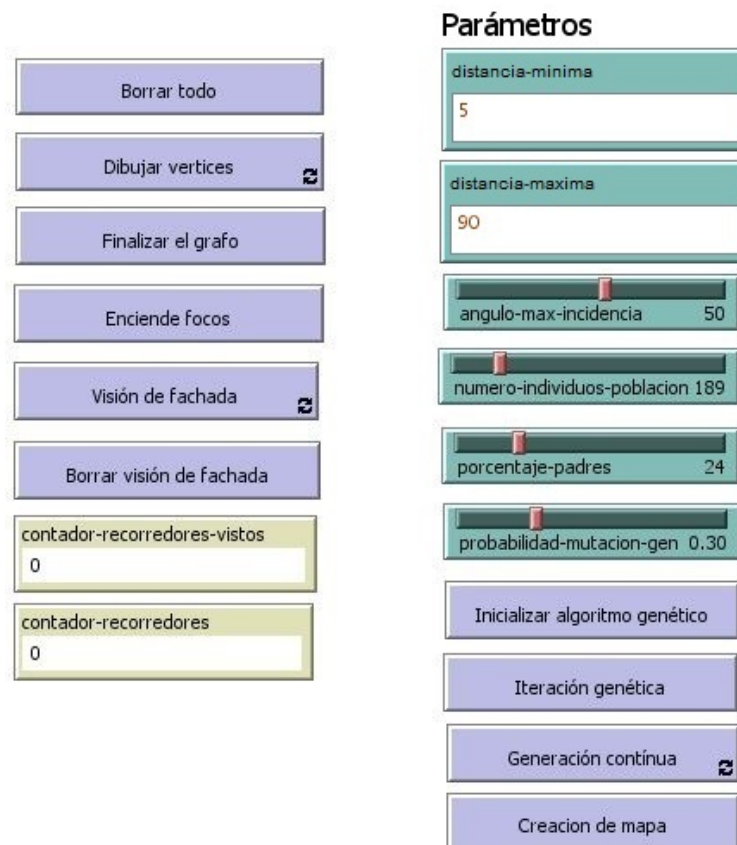


Figura 5.44: Interfaz del modelo diseñado en Net-Logo. Elaboración Propia.

Paso 3. Introducir el valor de  $\gamma_{maxadm}$  → Utilizar el deslizador «angulo-max-incidencia». Es el ángulo máximo admisible que puede formar el vector normal en la superficie de un punto leído con el rayo de visión, puede ir de 0 a 90°. Se ha realizado un estudio acerca del valor máximo que puede tomar este ángulo, el cual se detalla en el Apéndice B de esta memoria de tesis.

Una vez definidos los parámetros de funcionamiento del instrumento a utilizar, se continúa con la definición del polígono que se desea vigilar, el cual representa el recinto exterior del edificio en planta.

Paso 4. Pulsar sobre el botón «Borrar todo».

Paso 5. Pulsar sobre el botón «Dibujar vértices».

Paso 6. Introducir los puntos que define los vértices del polígono que se desea

vigilar.

Se debe tener en cuenta que en función al sentido con el que se dibuje, se procede a vigilar desde un lado u otro del polígono. En el caso de dibujar en sentido horario, se vigila desde el lado externo al polígono. En el caso de dibujar los vértices en sentido antihorario, se vigila desde el lado interior del polígono (Véase Figura 5.45).

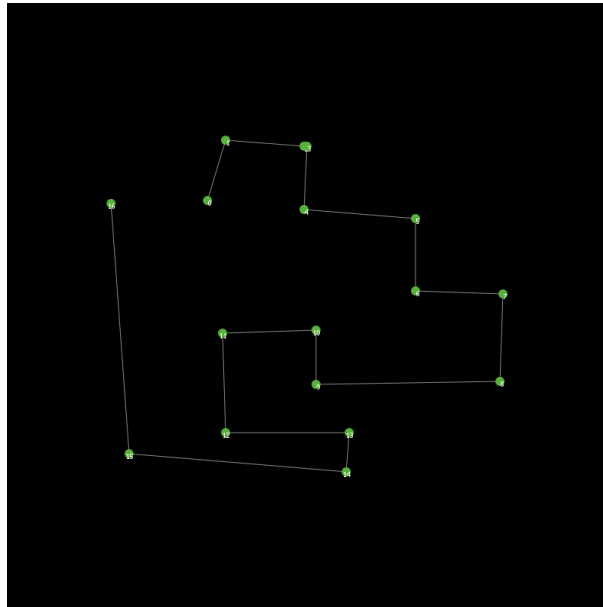


Figura 5.45: Dibujar los vértices en el sentido deseado. Elaboración Propia.

Paso 7. Pulsar el botón «Finalizar grafo».

De esta manera se colorea de blanco aquella zona en la que no se van a colocar los componentes de los individuos, y quedando así el mundo dividido en tres partes: la blanca, la naranja y la negra. La zona negra es aquella desde la cual se va a realizar la vigilancia del polígono (Véase Figura 5.46).

Paso 8. Pulsar el botón «Enciende focos».

En este paso se consigue el terreno inspeccionado. Desde cada *patch* que compone el borde del polígono es lanzada un área de visión que determina la zona desde la cual ese *patch* es correctamente vigilado, atendiendo a la distancia máxima,  $D$ , a la distancia mínima,  $d$  y al ángulo máximo admisible,  $\gamma_{maxadm}$  (Véase Figura 5.47).



Figura 5.46: Separación del espacio en tres regiones: interior del polígono, exterior del polígono y borde del polígono. Elaboración Propia.

En el código adjunto en el Apéndice D, el ángulo  $\gamma_{maxadm}$  se encuentra fijado en  $70^\circ$ .

Paso 9. (OPCIONAL) Pulsar el botón «Visión fachada».

En este paso, se trata de identificar de forma manual qué parte de una fachada es vigilada satisfactoriamente desde un punto concreto. Se trata de un paso que no es necesario, pero que resulta de gran utilidad para tener una idea del alcance con el que se está trabajando, o para comprobar si una toma de datos se ha realizado de forma correcta.

- a) Seleccionar un punto cualquiera del exterior del polígono.  
De esta forma se localiza la zona que es correctamente visualizada, quedando sombreada de color azul (Véase Figura 5.48).  
En la casilla de salida de datos «contador-recorredores-vistos» se suman los *patches* de la fachada que están siendo vigilados.
- b) Pulsar el botón «Borrar visión de fachada» para volver al estado inicial, en el cual se encuentra el terreno inspeccionado, pero sin ningún punto de vista seleccionado.

Paso 10. Fijar los parámetros del Algoritmo Genético.



Figura 5.47: Terreno inspeccionado. Elaboración Propia.

- a) Fijar el Tamaño de la Población Inicial a través del deslizador «numero-individuos-poblacion».
- b) Fijar la Probabilidad de Cruce a través del deslizador «porcentaje-padres».
- c) Fijar la Probabilidad de Mutación a través del deslizador «probabilidad-mutacion-gen».

Paso 11. Pulsar el botón «Inicializar algoritmo genético».

Paso 12. (OPCIONAL) Pulsar el botón «Iteración genética».

Este paso refleja una iteración del AG.

Paso 13. Pulsar el botón «Generación continua».

El AG comienza a crear generaciones de forma automática, atendiendo a la Función Objetivo y a los valores de entrada proporcionados. En la pantalla de *Net-Logo* se visualiza el mejor individuo de cada generación.

Paso 14. Pulsar de nuevo el botón «Generación continua».

De esta forma se paraliza el AG, obteniéndose una solución que puede considerarse óptima. En la Figura 5.49 se muestra una solución obte-

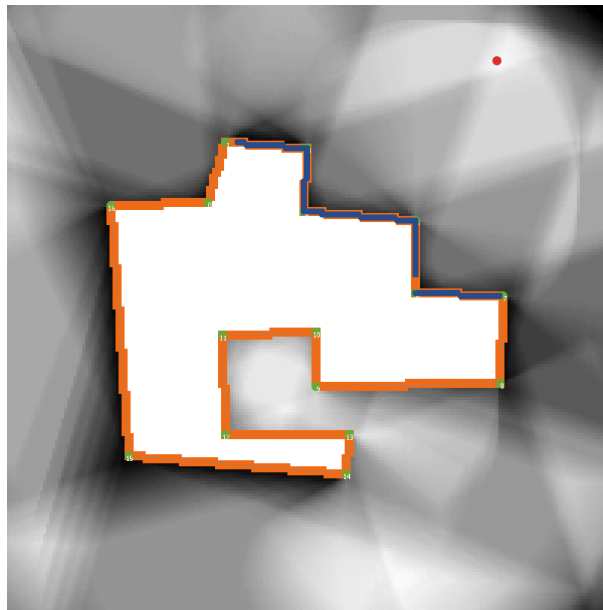


Figura 5.48: Opcional: Pinchar sobre posibles puntos de vista para analizar la visibilidad de los mismos. Elaboración Propia.

nida para el ejemplo utilizado. En la casilla «contador-recorredores-vistos» el usuario puede comprobar si se ha alcanzado la vigilancia completa del polígono.

El modelo exporta los datos de las distintas iteraciones del AG a un archivo de extensión .CSV, en el cual es posible analizar la evolución del Valor de Ajuste, así como la evolución del número de componentes del mejor individuo de cada generación, y las coordenadas cartesianas de los mismos. Los datos exportados a este fichero son, por cada iteración, los siguientes: el valor de ajuste del mejor individuo, el número de la iteración, el tiempo de ejecución, las coordenadas de los *patches* de los componentes que conforman al mejor individuo.

### 5.3.5. Resultados

Una vez expuesto el proceso por el cual se ha llevado a cabo el diseño de este segundo método basado en técnicas metaheurísticas, se dispone a analizar algunos de los resultados obtenidos. Para ello, diferentes formas poligonales han sido introducidas en el modelo, y se ha analizado la eficacia del Algoritmo Genético diseñado.



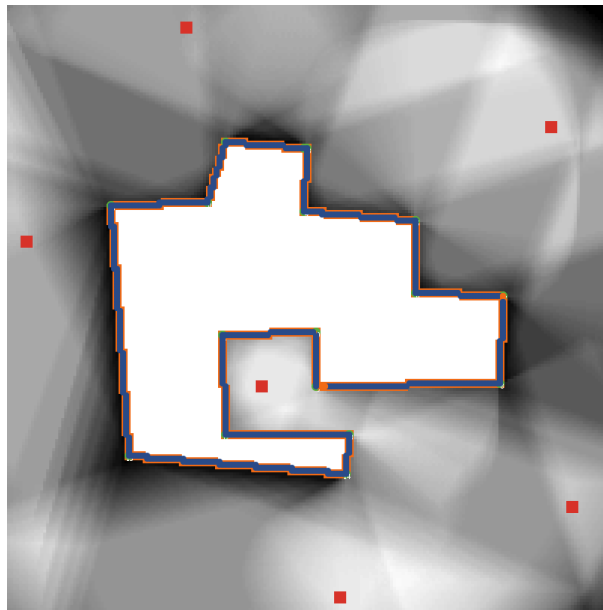


Figura 5.49: Solución obtenida a través del AG. Con seis posicionamientos es posible cubrir la vigilancia completa del polígono. Elaboración Propia.

En cada uno de los casos que se exponen a continuación han variado los siguientes factores:

- Tamaño del mundo.
- Rango de trabajo del instrumento.
- $\gamma_{maxadm}$ .
- Recorredores a vigilar según la forma del polígono introducido.
- Sentido de la introducción de los vértices, pudiendo variar entre horario o antihorario.
- Número de individuos de la población.
- Porcentaje de padres.
- Probabilidad de mutación.

#### 5.3.5.1. Caso I

En este caso, los datos introducidos han sido los siguientes:

- Tamaño del mundo: 200x200 *patches*.
- Rango de trabajo del instrumento: 5-30.
- $\gamma_{maxadm}$ : 70°.
- Recorredores a vigilar según la forma del polígono introducido: 615.
- Sentido de la introducción de los vértices: horario.

Tras la inspección del terreno las zonas quedan coloreadas como se muestra en la Figura 5.50.

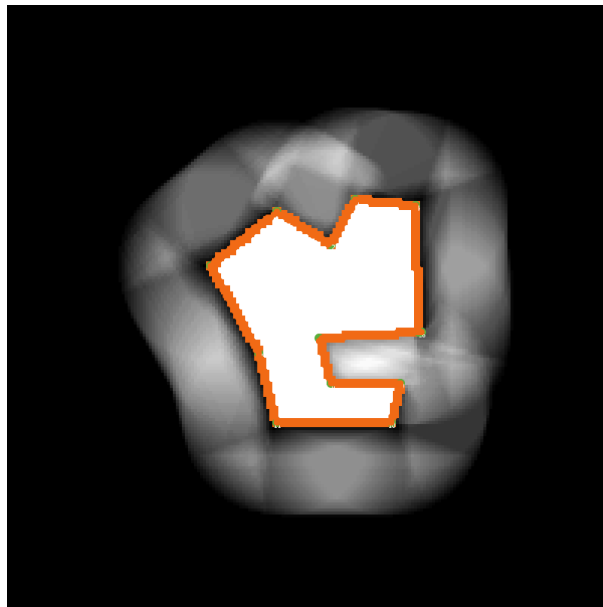


Figura 5.50: Terreno inspeccionado en Caso I. Elaboración Propia.

Tras ello se inicializa el Algoritmo Genético con los siguientes parámetros:

- Número de individuos de la población: 100.
- Porcentaje de padres: 10.

- Probabilidad de mutación: 0,25.

El resultado obtenido es el que se muestra en la Figura 5.51, dando como resultado un total de siete posicionamientos para vigilar de forma correcta el polígono dado.

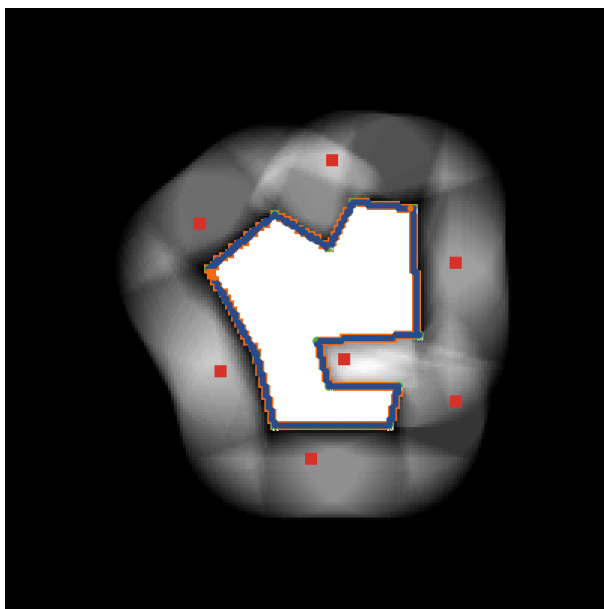


Figura 5.51: Solución obtenida tras aplicar el AG, Caso I. Elaboración Propia.

La gráfica donde se refleja la evolución del Valor de Ajuste (*Fitness*) y la evolución del número de componentes necesarios se muestra en la Figura 5.52. En el eje de abscisas se representan las iteraciones ejecutadas por el algoritmo, aproximadamente 5000; en el eje de ordenadas primario se representa el Valor de Ajuste, el cual es un valor que tiende a maximizarse, sin que exista un tope; por último, en el eje de ordenadas secundario se representa el valor del número de componentes calculados para visualizar el objeto. Puede observarse cómo mientras que el Valor de Ajuste (o *Fitness*) aumenta, el número de componentes disminuye.

#### 5.3.5.2. Caso II

De la misma forma que se ha procedido en el Caso I, se realiza una segunda prueba con otra forma poligonal distinta.

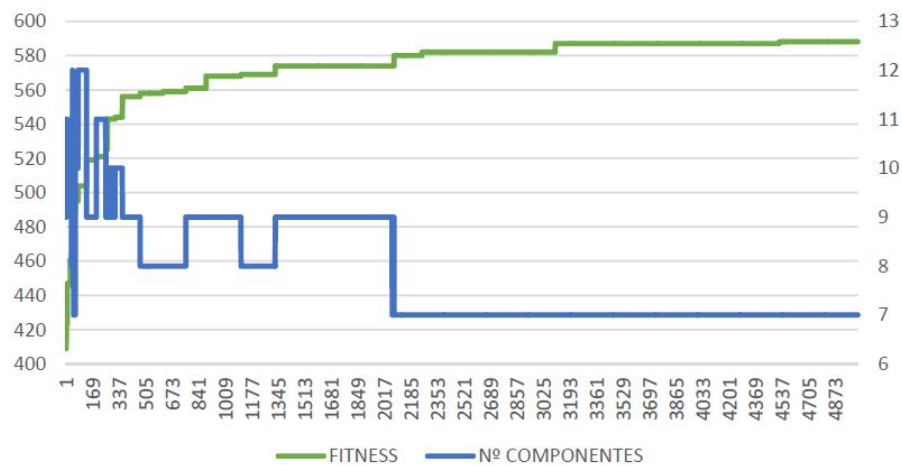


Figura 5.52: Evolución del Valor de Ajuste y del número de componentes, Caso I. Elaboración Propia.

En este segundo caso, los datos de entrada son los siguientes:

- Tamaño del mundo: 200x200 *patches*.
- Rango de trabajo del instrumento: 5-60.
- $\gamma_{maxadm}$ : 70°.
- Recorredores a vigilar según la forma del polígono introducido: 802.
- Sentido de la introducción de los vértices: horario.

Una vez inspeccionado el terreno, se ejecuta el Algoritmo Genético con los siguientes parámetros:

- Número de individuos de la población: 189.
- Porcentaje de padres: 20.
- Probabilidad de mutación: 0,30.

El resultado obtenido tanto de la inspección del terreno, como de la aplicación del AG es el que aparece en la Figura 5.53, obteniéndose un total de ocho puntos de vista.

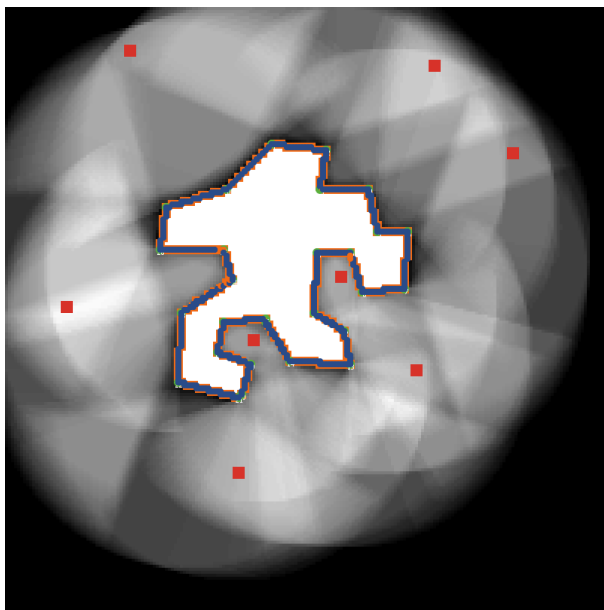


Figura 5.53: Solución obtenida tras aplicar el AG, Caso II. Elaboración Propia.

La gráfica en la que se muestra la evolución del Valor de Ajuste así como del número de componentes necesarios a lo largo de 923 iteraciones, es la que se refleja en la Figura 5.54.

#### 5.3.5.3. Caso III

En este otro ejemplo, los datos de entrada son los siguientes:

- Tamaño del mundo: 200x200 *patches*.
- Rango de trabajo del instrumento: 5-90.
- $\gamma_{maxadm}$ : 70°.
- Recorredores a vigilar según la forma del polígono introducido: 1398.
- Sentido de la introducción de los vértices: horario.

Una vez inspeccionado el terreno, se ejecuta el Algoritmo Genético con los siguientes parámetros:

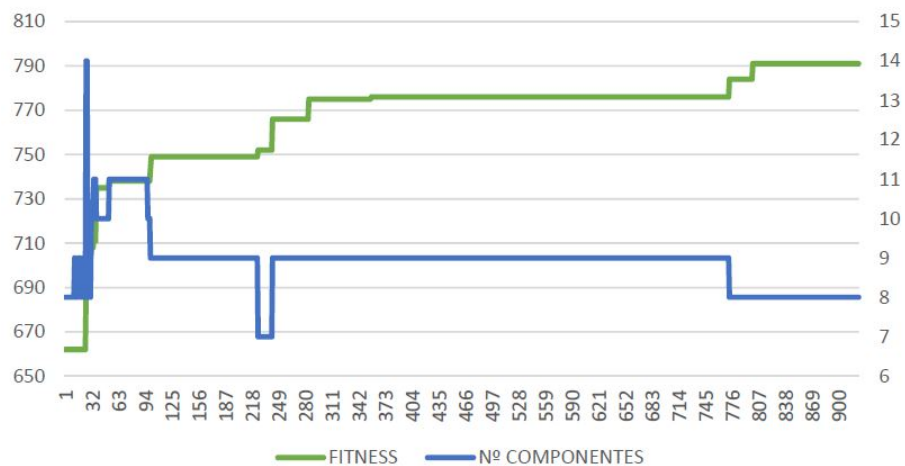


Figura 5.54: Evolución del Valor de Ajuste y del número de componentes, Caso II. Elaboración Propia.

- Número de individuos de la población: 189.
- Porcentaje de padres: 20.
- Probabilidad de mutación: 0,30.

En la Figura 5.55 puede verse la forma del polígono introducido, el terreno inspeccionado, así como los puntos de vista obtenidos.

La gráfica correspondiente es la que se refleja en la Figura 5.56, a lo largo de 703 iteraciones.

#### 5.3.5.4. Caso IV

Por último, para este caso se han introducido los vértices del polígono en sentido antihorario, por lo que el modelo va a buscar puntos de vista en el interior del polígono.

Los datos de entrada son los siguientes:

- Tamaño del mundo: 300x300 *patches*.
- Rango de trabajo del instrumento: 10-50.

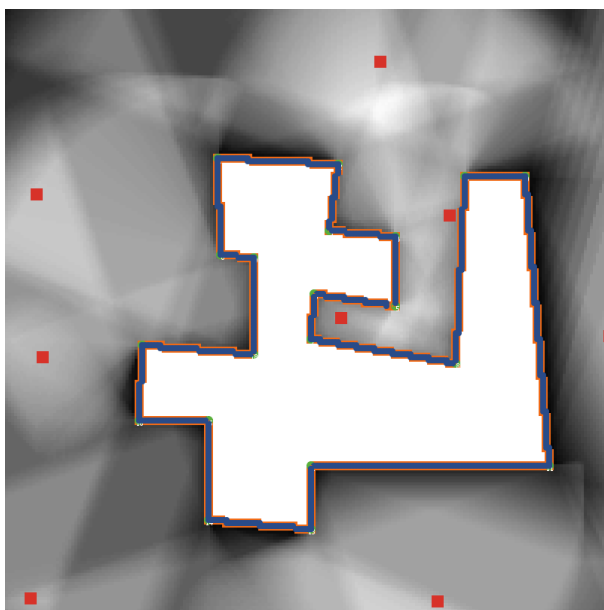


Figura 5.55: Solución obtenida tras aplicar el AG, Caso III. Elaboración Propia.

- $\gamma_{maxadm}$ :  $70^\circ$ .
- Recorredores a vigilar según la forma del polígono introducido: 1820.
- Sentido de la introducción de los vértices: antihorario.

Una vez inspeccionado el terreno, se ejecuta el Algoritmo Genético con los siguientes parámetros:

- Número de individuos de la población: 152.
- Porcentaje de padres: 71.
- Probabilidad de mutación: 0,28.

El resultado obtenido es el que se muestra en la Figura 5.57, obteniéndose once puntos de vista.

Para concluir con este apartado se muestra el gráfico que refleja la evolución del AG (Véase Figura 5.58).

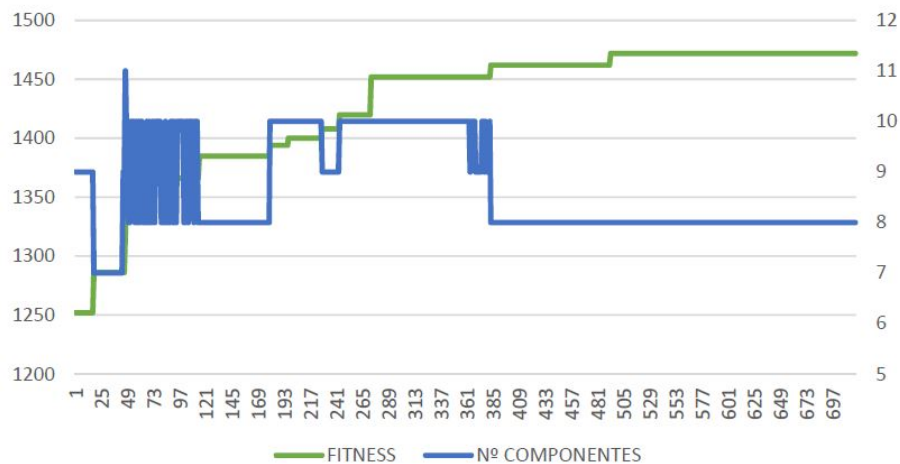


Figura 5.56: Evolución del Valor de Ajuste y del número de componentes, Caso III. Elaboración Propia.

Para poner en funcionamiento el modelo descrito en la Sección 5.3, es necesario instalar en un equipo la plataforma *Net-Logo*, disponible de forma gratuita en <https://ccl.northwestern.edu/netlogo/>, (Wilensky, 1999). La última versión es la 6.0.1.

La interfaz de esta plataforma se divide en tres pestañas: Ejecutar, Información y Código.

En la pestaña «Ejecutar» deben estar creados todos los botones y deslizadores que conforman la interfaz del modelo. En la pestaña «Información» se puede encontrar información sobre el modelo. En la pestaña «Código» aparece el código escrito en el Apéndice D.

En la documentación digital anexa a esta memoria de tesis puede encontrarse el archivo de extensión *.nlogo*, la cual se encuentra preparada para ser ejecutada siguiendo los pasos descritos en la Sección 5.3.4.

### 5.3.6. Conclusiones

Tras los distintos casos estudiados, se puede verificar el correcto funcionamiento de la aplicación diseñada. La Función Objetivo es capaz de minimizar el número de componentes que vigila el polígono, mientras que maximiza el área vigilada, todo ello cumpliendo una serie de restricciones dadas.



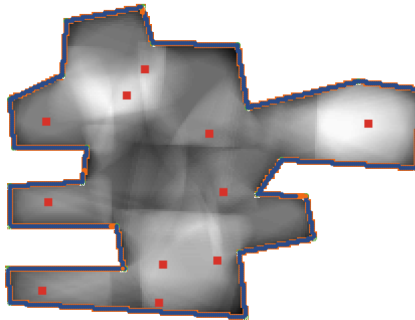


Figura 5.57: Solución obtenida tras aplicar el AG, Caso IV. Elaboración Propia.

Además, cabe destacar que el modelo diseñado ya es útil de por sí sin necesidad de aplicar el Algoritmo Genético, pues con el modo manual de Visión de fachada se puede decidir desde qué puntos de vista ejecutar la toma de datos geométricos.

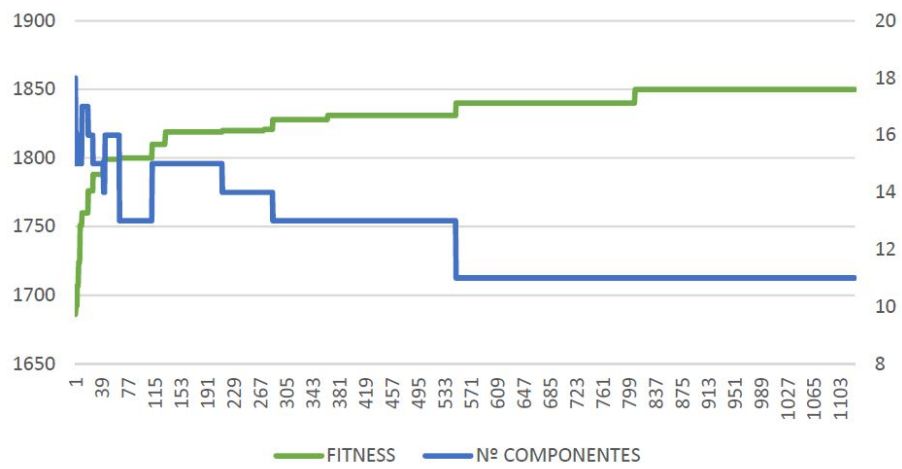


Figura 5.58: Evolución del Valor de Ajuste y del número de componentes, Caso IV. Elaboración Propia.

## Capítulo 6

# Resultados

En el presente trabajo de investigación se han obtenido dos herramientas que sirven a los profesionales de la Topografía, la Ingeniería y la Arquitectura para la toma de decisiones con respecto a qué puntos de vista escoger para garantizar una correcta toma de datos geométricos de un edificio. Para diseñar estos métodos, ha sido necesario realizar un estudio acerca de las técnicas de optimización más adecuadas.

El primer método se trata de un Algoritmo para hallar las áreas de visibilidad suficientes desde las cuales vigilar el borde de un polígono desde el lado exterior al mismo, descrito en la Sección 5.1. Se trata de un algoritmo determinista basado en el Problema de la Fortaleza, el cual es una de las variantes del Problema de la Galería de Arte.

Con esta primera herramienta se hallan las áreas de visibilidad suficientes para visualizar de forma completa una forma poligonal, la cual representa el recinto exterior edificado en planta, del cual se desea realizar un levantamiento arquitectónico (Apartado 5.1.2). Este algoritmo es aplicable a cualquier tipo de polígono.

No obstante, se ha obtenido una adaptación de este método para el caso especial de aquellos edificios que pueden ser representados por plantas representadas por polígonos ortogonales (Apartado 5.1.3). En este tipo de polígonos el número de posicionamientos desde los cuales realizar la toma de datos se reduce.

La herramienta se caracteriza por su sencillez, pudiéndose realizar el cálculo de estas zonas de visibilidad de forma manual, sin precisar de una aplicación

informática que ejecute el algoritmo diseñado, pues éste es sencillo y no precisa de cálculos de alta complejidad computacional. No obstante, se ha realizado la implementación de este método como una aplicación web en *PYTHON*, la cual queda descrita en el Apartado 5.1.4. El código puede encontrarse en la documentación digital anexa a esta memoria de tesis.

Tras analizar distintos casos de aplicación de la aplicación desarrollada a formas poligonales de diversos tipos en el Apartado 5.1.5, se obtienen resultados favorables, pues de forma rápida y sencilla se consigue determinar un número reducido de áreas de visibilidad que permiten la correcta visualización del borde del polígono desde el lado exterior al mismo.

Por último, con respecto al primer método desarrollado cabe destacar que éste resulta además útil para cualquier trabajo de optimización en los que se desee vigilar un objeto con el mínimo número de posicionamientos posibles.

Por otro lado, para diseñar la segunda herramienta se ha conseguido implementar un Algoritmo Genético a través de un modelo multiagente, el cual permite calcular un conjunto de puntos de vista que puede ser considerado como óptimo para visualizar de forma completa un edificio, desde el lado exterior al mismo (Sección 5.3). En este segundo método, el edificio es también representado por el polígono que define el recinto exterior edificado en planta.

El modelo diseñado ofrece la posibilidad de introducir las variables que se crean convenientes: rango de trabajo del instrumento, ángulo mínimo de incidencia para la correcta visualización, etc. Lo cierto es que al ofrecer la posibilidad de modelar con tantas restricciones como se crea conveniente, la aplicación se encuentra en un proceso continuo de mejora e introducción de nuevas restricciones.

Este modelo no solo sirve para ejecutar un AG, sino que también es útil para conocer la parte del polígono que es visualizada desde un punto concreto.

Tras su aplicación a distintos casos (Apartado 5.3.5) se observa que la aplicación responde de forma adecuada, adaptándose a las diferentes casuísticas planteadas, incluso siendo útil para calcular los puntos de vista situados en el interior del recinto edificado, para visualizar el borde del mismo.

El código se encuentra escrito en el Apéndice D, y el modelo ejecutable está disponible en la documentación digital anexa a esta memoria de tesis.

## Capítulo 7

# Conclusiones

Después de la amplia revisión realizada en cuanto a las publicaciones referentes a la planificación para la toma de datos geométricos en levantamientos arquitectónicos, se concluye que son muy escasos los trabajos dedicados a establecer previamente al levantamiento los puntos idóneos para realizar una toma de datos eficaz.

Una vez dicho esto, se llega a la conclusión de que el diseño de un método que sirva de apoyo a los profesionales de la Topografía, resulta necesario.

Por ello, se han diseñado dos métodos, los cuales son bastante distintos entre sí. Estos dos métodos se diferencian ya que cada uno de ellos están diseñados basándose en distintas técnicas de optimización.

Para ambos métodos es necesario partir de una planta aproximada previa del recinto del edificio sobre el que se desea realizar el trabajo de levantamiento arquitectónico. Esta necesidad podría plantear al lector la paradoja de necesitar una representación del objeto para obtener la representación del objeto. Pero obviamente la nueva representación que se busca es más completa, precisa y exhaustiva, mientras que la planta de la que se parte es aproximada. Esta situación es común a estas propuestas y las existentes en la epistemología sobre el tema.

De esta forma se parte de una planta de baja precisión, a partir de la cual se realizan las pruebas necesarias. La validez de los métodos está estrechamente relacionada con la calidad y precisión de la planta previa de la que se parte.

El primer método consta de un algoritmo determinista, el cual es el más

sencillo de los dos, tanto en su planteamiento como en su desarrollo (Sección 5.1). A través de este algoritmo se encuentran las áreas de visibilidad suficientes para garantizar la correcta visibilidad de un edificio. En este caso el edificio se ha reducido a una forma poligonal, la cual representa la planta del recinto exterior del edificio. Pueden destacarse las siguientes conclusiones de este método:

- Se trata de un método sencillo.
- Garantiza la vigilancia completa de una forma poligonal desde el lado exterior a la misma.
- Sirve como ayuda para estimar un número suficiente de posicionamientos.
- Este algoritmo proporciona las áreas de visibilidad suficientes para vigilar un edificio, tratando a todos los puntos pertenecientes a un mismo área de igual forma.
- Las áreas de visibilidad son creadas partiendo de las aristas del polígono que define la planta del edificio. Por lo que una fachada es considerada una entidad indivisible. Esto quiere decir que las áreas se forman para vigilar una fachada de forma completa, y no por tramos de la misma.
- La implementación se ha realizado de forma satisfactoria, ejecutándose en un corto periodo de tiempo y dando respuesta a cualquier polígono introducido, independientemente de su complejidad.

Pasando al segundo método diseñado, cabe destacar que éste ha pasado por distintas fases hasta su estado actual, y que sigue en proceso de mejora continua. El dominio de búsqueda de solución ha ido ampliándose en las sucesivas versiones.

En un principio se estudia el diseño de un AG cuyos posicionamientos fueran situados sobre las áreas proporcionadas por el método anterior. Sin embargo, esta manera de diseñar un AG no es la más adecuada, ya que uno de los principios de funcionamiento de la computación evolutiva consiste en dejar que por sí mismo encuentre una solución óptima, dándole suficientes grados de libertad para ello; de otra forma el algoritmo convergería o se estancaría rápidamente.

---

El modelo presentado supone un importante paso y una herramienta útil para ejecutar un trabajo de levantamiento arquitectónico, no solo para la planificación de los puntos óptimos de estacionamiento sino también para comprobar si las estaciones empleadas en un levantamiento son suficientes para el registro de la geometría completa del edificio.

Las conclusiones obtenidas de este segundo método pueden resumirse en las siguientes:

- La decisión sobre qué puntos escoger para estacionar deja de ser una operación únicamente basada en la experiencia y en el buen hacer de los profesionales.
- El modelo diseñado ha resultado sorprendentemente versátil y flexible, adaptándose a las restricciones impuestas y respondiendo de forma adecuada.
- En el modelo presentado no se consideran a las aristas del polígono que define la forma de la planta del edificio, como una entidad indivisible. Esto aporta una gran ventaja con respecto al primer método expuesto, ya que es común que ocurra que para vigilar una única fachada sea necesario utilizar más de un posicionamiento, como puede ocurrir si, por ejemplo, la longitud total de la fachada fuera mayor al alcance máximo del aparato, o los ángulos de incidencia de las visuales a la fachada estuviesen por debajo de los parámetros establecidos.
- El modelo diseñado también sirve para localizar los puntos de vista desde los cuales realizar la toma de datos geométricos para levantar un recinto edificado desde el interior del mismo.
- Por otro lado, es importante resaltar la posibilidad que ofrece el modelo de importar datos creados a partir de programas GIS (*Geographic Information System*).

Con todo esto, no se deben perder de vista muchas mejoras de los procedimientos planteados, y que se esperan poder abordar en futuros trabajos. Se hablará de ellas en el Capítulo 8.

Como conclusión común sobre los métodos diseñados y desarrollados en este trabajo de investigación es importante mencionar que estos métodos pueden aplicarse a otras disciplinas que no tienen porqué ser la Topografía y el levantamiento de bienes arquitectónicos. Pues también pueden servir para dar

respuesta a problemas de vigilancia o de cobertura, como por ejemplo sensores, redes *Wi-Fi*, cámaras de seguridad o trabajos de inspección.



## Capítulo 8

# Líneas futuras de investigación

- Las mejoras asumibles en el modelo multiagente actual de *Net-Logo* son las siguientes:
  - Consideración del solape de datos leídos desde distintos puntos de vista. Se debe tener en cuenta que los distintos puntos de vista compartan un tanto por ciento de visibilidad común con algún otro punto de vista, de forma que la unión de las distintas nubes de puntos sea más sencilla.
  - Optimización en el camino a recorrer por la persona encargada de ejecutar el levantamiento arquitectónico, favoreciendo a aquellas soluciones cuyo camino a recorrer sea el mínimo posible.
  - Posibilidad de importar y exportar datos georreferenciados a través de GIS.
  - Consideración de los elementos externos al edificio que puedan dificultar la correcta visibilidad a través de ellos. Por ejemplo, en el caso de existir edificios alrededor, o bien, elementos verticales como vegetación o instalaciones las cuales impiden que se pueda visualizar a través de ellos.
  - Consideración de zonas sobre las que no es posible estacionar, pero que no limiten la visibilidad a través de ellas. Es el caso de zonas en las que exista agua, tráfico, o una propiedad a la que no es posible acceder, pero que sin embargo no impide que se pueda observar a través de ellas.
- Implementación en JAVA del método basado en técnicas metaheurísticas con un enfoque alternativo a los sistemas multiagentes.

- Comparación con otras técnicas metaheurísticas, como por ejemplo, la técnica de Optimización por Enjambre de Partículas, siguiendo los pasos del investigador Hossam El-Din Fawzy <sup>1</sup>, quien realizó una comparación entre la técnica del Algoritmo Genético, y la técnica de Optimización por Enjambre de Partículas, para el diseño de redes de fotogrametría de objeto cercano (Fawzy, 2015).
- Modelado del problema en tres dimensiones, considerando así:
  - Las oclusiones producidas por la propia geometría del edificio: vuelos, abultados, cornisas, retranqueos, etc.
  - La optimización de recorrido para toma de datos con drones.

---

<sup>1</sup>Departamento de Ingeniería Civil. Facultad de Ingeniería. Universidad de Kafr El-Sheikh. Kafr El-Sheikh, Egipto

## Apéndice A

### Estudio de caso

Tras haber expuesto de forma detallada el Método de Optimización en el Posicionamiento para la realización de un Levantamiento Arquitectónico, basado en técnicas metaheurísticas, desarrollado en la Sección 5.3, en este apéndice se describe la puesta en práctica sobre un edificio existente.

El edificio elegido para la aplicación de la metodología desarrollada es el Molino de Marea del Río Arillo, situado en el límite entre los municipios de Cádiz y San Fernando.

En este apartado, en primer lugar se realiza una introducción acerca de la importancia que tuvieron los molinos de marea durante varios siglos, y se describe de forma breve su funcionamiento. En segundo lugar, se describe el entorno en el que se encuentra el Molino de Marea del Río Arillo, realizando posteriormente un pequeño análisis de este edificio concreto. Por último, se describe el proceso de trabajo realizado de planificación y trabajo de campo para el escaneo de este edificio.

Para este trabajo ha sido fundamental el libro *Molinos de Marea de la Bahía de Cádiz (Siglos XVI-XIX)* obra del gaditano Julio Molina Font (Molina Font, 2001), así como el trabajo realizado por la arquitecta gaditana Carmen Machuca (Machuca Macías, 2013) y el libro *La Casa Salinera* de Juan Manuel Suárez Japón (Suárez Japón, 1989).

## **A.1. Los Molinos de Marea**

A partir del siglo XIII comenzaron a construirse distintos tipos de molinos hidráulicos en la Península Ibérica. Entre ellos se encontraban los molinos de mar o mareales.

Este tipo de construcción aprovecha la fuerza de las mareas como fuente de energía necesaria e inagotable para hacer girar las piedras de molienda. Normalmente, se usaban para moler trigo u otro tipo de cereal.

Era frecuente encontrar agrupaciones de ingenios hidráulicos. Podían encontrarse en la Bahía de Cádiz, en el litoral onubense, en las costas gallegas, Asturias, País Vasco, Cantabria, así como en el Algarve y en los alrededores de Lisboa.

No sólo fue un tipo de construcción común en la Península Ibérica, más de medio millar de molinos de marea ocuparon el litoral atlántico de Irlanda, Gran Bretaña, Países Bajos, Bélgica, Francia, España y Portugal desde la Edad Media (Véase Figura A.1).

### **A.1.1. Desarrollo histórico**

Aunque los molinos de marea comenzaron a construirse desde el siglo XIII, es entre los siglos XVII y XVIII cuando llegan a su máximo apogeo. Permitían una actividad industrial modélica, no contaminante y que utilizaban un recurso energético gratuito e inagotable. Eran edificaciones que abastecían de forma sostenible a pequeñas comunidades, quedando integradas en el paisaje, y con un impacto ambiental insignificante.

Poco después, debido a la revolución industrial, estas factorías comenzaron su declive, a finales del siglo XIX. Desgraciadamente, la mayoría de los molinos de marea se encuentran en estado ruinoso. Algunos, sin embargo, han sido rehabilitados para usos turísticos o como centros de interpretación del entorno en el que se encuentran.

## A.1. LOS MOLINOS DE MAREA

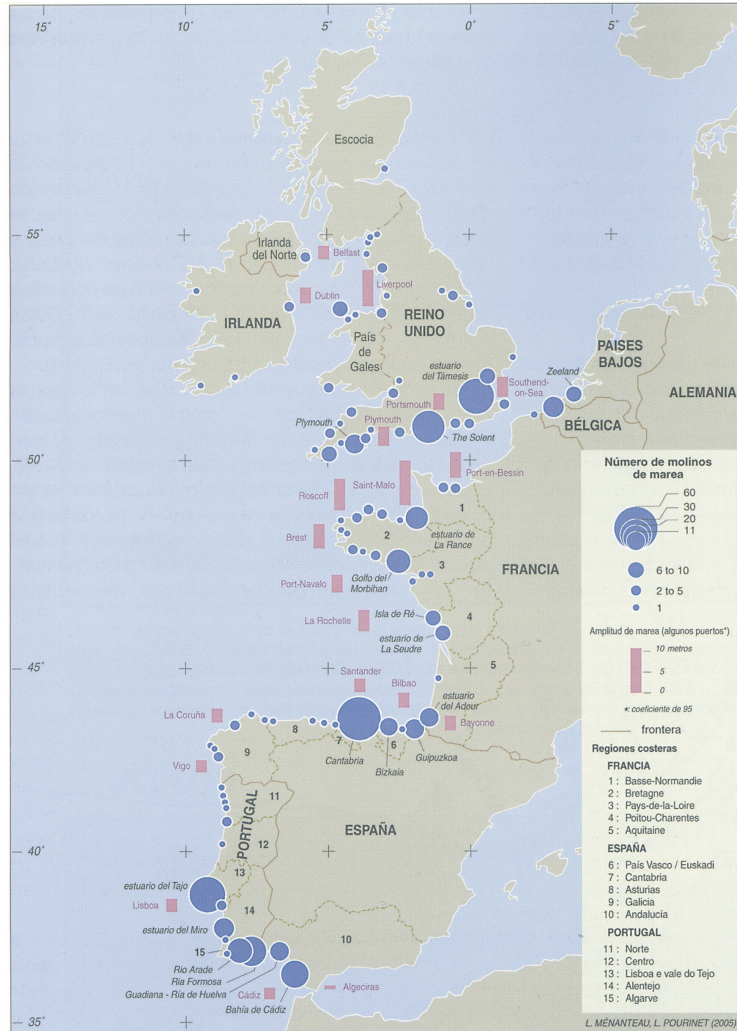


Figura A.1: Distribución geográfica de los molinos de marea (Ménanteau and Pourinet, 2005).

### A.1.2. Funcionamiento

Como se ha mencionado anteriormente, este tipo de molinos aprovecha la fuerza inagotable del mar para moler cereales, generalmente trigo. Gracias a la subida de la marea o pleamar, se conseguía almacenar una gran cantidad de agua en un estanque, denominado caldera o presa. La presión del agua embalsada era capaz de cerrar unas compuertas basculantes. Una vez pasadas dos horas después de la pleamar, la marea habría bajado lo suficiente como para dejar salir el agua almacenada a través de las cárcavas (Véase Figura A.2).

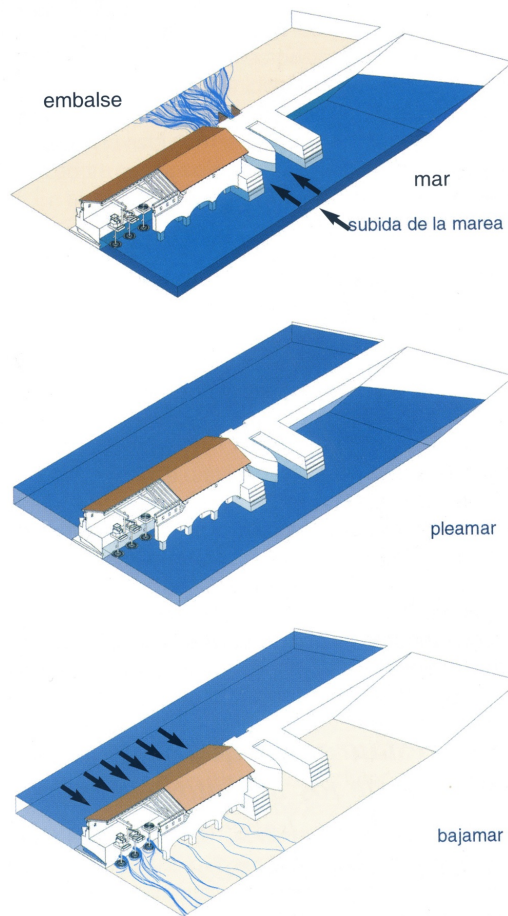


Figura A.2: Esquema de funcionamiento de un molino de marea (Azurmendi Pérez, 2011).

Esto se conseguía mediante el izado a mano de las compuertas, y permitiendo que el agua entrara a presión a través de los saetines. Al incidir el agua sobre las aspas del rodete, se provocaba el giro de las ruedas motrices del molino. Éstas a su vez transmitían el giro a las piedras de moler (Molina Font, 2001).

Este trabajo variaba de horario cada día, pues el ciclo de mareas es de doce horas. Las pleamares se desplazan una hora cada día, y los coeficientes de marea van variando a lo largo del mes. De lo que se puede deducir que el trabajo del campesino estaba sujeto a estos horarios.

## A.2. El Molino del Río Arillo y su entorno

### A.2.1. El Parque Natural de la Bahía de Cádiz

El molino sobre el cual se ha aplicado el método diseñado, se encuentra ubicado en el Parque Natural de la Bahía de Cádiz.

Este parque tiene una extensión aproximada de 10000 hectáreas, configuradas por fangos intermareales, salinas y marismas, formándose un entramado de caños, canales y brazos por los cuales el agua de las mareas circula, alimentando salinas, esteros y lucios (Molina Font, 2001). Está situado entre el Parque Nacional de Doñana, y el estrecho de Gibraltar, en la costa gaditana occidental, comprendiendo los municipios de Cádiz, Chiclana de la Frontera, El Puerto de Santa María, Puerto Real y San Fernando (Véase Figura A.3).

Este paisaje se caracteriza por su variedad cromática de tonos verdes, marrones y azules, así como por la geometría singular y laberíntica de sus canales y brazos (Véanse Figuras A.4 y A.5). «Las texturas dominantes corresponden al agua y la tierra, matizadas según el nivel hídrico de los caños de acuerdo a la marea y a la proporción agua/tierra, dependiente del estado de aterramiento» (Machuca Macías, 2013).

Otra singularidad de este espacio natural es su planicie y la escasez de masa forestal. Sin duda el protagonista es el mar, y todo gira a su alrededor: aves migratorias, fauna marina y actividad salinera. Esto es debido a que esta zona ha emergido con la bajada del nivel del mar, como queda reflejado en la Figura A.6.

El clima de esta zona es suave. La vegetación está adaptada a las condiciones saladas del terreno. En cuanto a la fauna, este espacio es uno de los más importantes a nivel europeo para las aves migratorias.

El Parque Natural de la Bahía de Cádiz es un espacio en continuo movimiento, como lo es la marea, que proporciona oxígeno y vitalidad a toda la marisma a través de los canales, que actúan como arterias de este organismo vivo. En este espacio no sólo es fundamental la energía solar para mantener vivo el ecosistema, sino que también es primordial la energía de la luna que mueve las mareas.

En el marco de la Bahía de Cádiz, se llegaron a construir más de veinte molinos de marea. Como no podía ser de otra forma, estas construcciones son, en su mayor parte, llevadas a cabo con el material autóctono por excelencia:



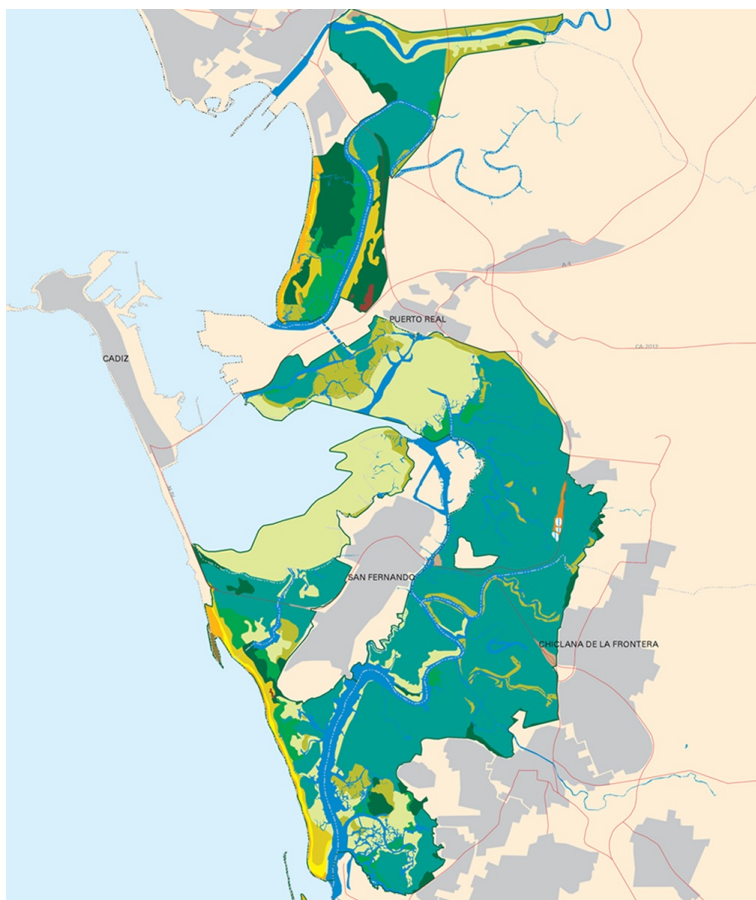


Figura A.3: Parque Natural de la Bah   de C  diz (Machuca Mac  as, 2013).

la piedra ostionera, la cual est   presente en la mayor  a de construcciones de esta zona, de todo tipo de usos y de todas las   pocas.

Durante siglos, la actividad salinera ha sido uno de los motores de esta zona. Aprovechando el potencial de esta tierra, la sal ha sido fuente de riqueza y trabajo para sus habitantes. Por ello, numerosas edificaciones singulares pueden encontrarse en este entorno, las cuales serv  an para llevar a cabo dicha actividad.

Como menciona la arquitecta gaditana Carmen Machuca «...la Bah  a de C  diz posee una innegable importancia hist  rica ligada a su privilegiada situaci  n geogr  fica y a sus caracter  sticas naturales. Esta condici  n motiv   desde tiempos remotos el establecimiento de sucesivas civilizaciones y culturas flo-



Figura A.4: Vista aérea del Parque Natural de la Bahía de Cádiz. Fuente (Instituto Geográfico Nacional, 2017).

recientes, que exploraron los valiosos recursos de la zona (pesca, puertos naturales, sal) y dejaron una importante huella cultural cuyos testimonios llegan hasta nuestros días» (Machuca Macías, 2013).

#### **A.2.2. Molinos de Marea de la Bahía de Cádiz**

La Bahía de Cádiz, configurada por marismas y laberintos arteriales de caños y esteros, fue un lugar idóneo para la instalación de este tipo de molinos. Para su ubicación se escogían caños propicios donde el agua del mar, por medio de la pleamar, llegara con facilidad.

El agua era almacenada en la presa o caldera. En muchas ocasiones, como en el caso del Río Arillo, la presa se construía aprovechando un ensanche del caño. En otros casos era construida de forma artificial. La capacidad de la presa era proporcional al número de piedras de molienda que poseía el molino. En la bahía podían encontrarse molinos con desde dos piedras hasta doce, como es el caso del molino del Río Arillo.

El molino se construía perpendicular al caño, cegándolo y dando paso al agua proveniente de la pleamar, a través de arcos y canales.

Este tipo de construcciones puede encuadrarse en la arquitectura popular. Eran construcciones muy adaptadas al medio, con similitudes con las casas salineras y ventorrillos de caminos. La cimentación se realizaba por medio de

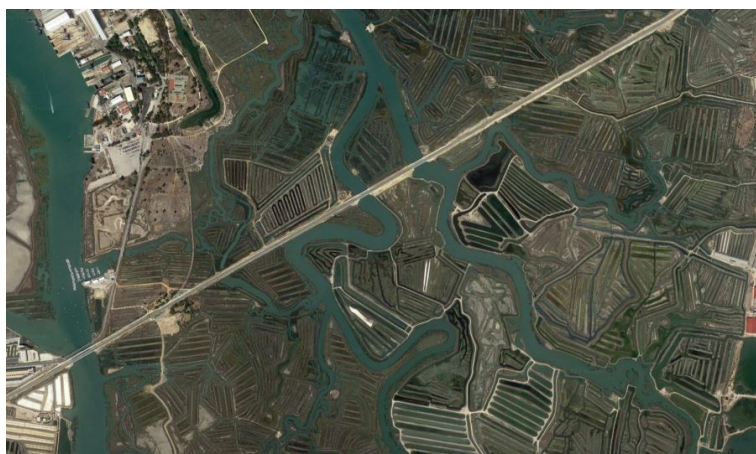


Figura A.5: Vista aérea del Parque Natural de la Bahía de Cádiz. Fuente (Instituto Geográfico Nacional, 2017).

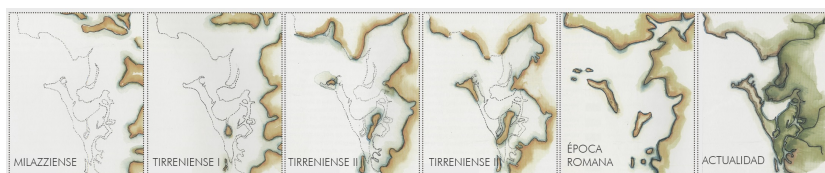


Figura A.6: Bajada del nivel del mar en la Bahía de Cádiz (Machuca Macías, 2013).

pilotes de madera de gran profundidad, al tratarse de un terreno fangoso. La madera de estos pilotes era generalmente de eucalipto.

Los materiales de construcción que se utilizaban iban desde las piedras a ladrillos o adobes. Se podían encontrar molinos con cubiertas a una sola agua, a dos aguas, e incluso con cubierta plana con azotea y con sus bordes adornados con remates geométricos. Aunque cada molino tuviera sus particularidades constructivas, todos compartían un factor común: la fábrica de los bajos del molino construida con piezas de cantería de piedra ostionera.

La piedra ostionera es una roca sedimentaria muy porosa formada por restos de conchas marinas y piedras erosionadas del mar. Es una piedra muy utilizada en Cádiz y en sus localidades cercanas. Es de color marrón y se caracteriza por contener incrustados trozos de crustáceos (Machuca Macías, 2013).

En la Figura A.7 se resumen las características de los molinos de marea de la Bahía de Cádiz. Por otro lado, en la Figura A.8 aparece reflejada la ubicación de los distintos molinos mareales que han existido en la Bahía de Cádiz.

Localización	Denominación	Siglo	Nº de piedras	Caño	Estado de conservación
Cádiz	Sierra	XVII			Desaparecido
Cádiz	Río Arillo	XVIII	12	Río Arillo	Muy deteriorado
Cádiz	Santibáñez	XV-XVI	3	Caño Roqueta	Muy deteriorado
Chiclana	Santa Cruz	XV	5	Caño Calera o Alcornocal	Bien conservado
Chiclana	Nuevo	XV	5		Algunos restos
Chiclana	Ormaza	XVIII	5	Caño del Molino	Desaparecido
Chiclana	Aceña			Río Iro	Desaparecido
Chiclana	De Barvitas	XVIII	5	Caño del Leñador	Bien conservado
El Puerto de Santa María	Molino del Caño	XIX	8	Caño del Molino	Bien conservado
Puerto Real	Guerra	XVI	9	Caño Felices	Desaparecido
Puerto Real	Molino de Triguero	XVI		Caño de la Marina	Desaparecido
Puerto Real	El Pilar	XVII	3	Caño Barbanera	Algunos restos
Puerto Real	De Ocio	XVIII	5	Caño Zurraque	Bien conservado
Puerto Real	Goyena	XVIII	6	Caño de la Marina	Muy deteriorado
Puerto Real	Molino de Mendoza			Caño de la Marina	Desaparecido
San Fernando	San José	XVI	8	Caño San José	Muy deteriorado
San Fernando	Fadrica	XVI		Fadrica	Desaparecido
San Fernando	Nuestra Señora de la Concepción	XVII		Brazo interior Río Arillo	Desaparecido
San Fernando	Saporito	XVIII	4	Caño Saporito	Bien conservado
San Fernando	Caño Herrera	XVIII	4	Caño Herrera	Muy deteriorado

Figura A.7: Resumen de características de los molinos de marea del Parque Natural de la Bahía de Cádiz. Elaboración propia. Fuente (Molina Font, 2001).

### A.2.3. El Molino de Marea del Río Arillo

El Molino de Río Arillo es considerado como uno de los molinos mareales más importantes de todo el occidente atlántico. Poseía doce piedras moledoras, y una gran capacidad de molienda. Tan sólo se conoce de la existencia de otros dos molinos de estas magnitudes: uno situado en Cantabria, con doce piedras; y otro situado en Portugal, con trece. Estuvo en funcionamiento hasta los años 30.

Esta edificación y su entorno fue declarada en 2002 Bien de Interés Cultural en el Catálogo General del Patrimonio Histórico Andaluz.

Además, no por todos es sabido que este molino alojaba en su interior el Salón del Obispo. Esto implica que este edificio no solo tenía uso industrial, sino que, dada su ubicación, era un edificio significativo para la ciudad de Cádiz.



## A.2. EL MOLINO DEL RÍO ARILLO Y SU ENTORNO



Figura A.8: Ubicación de los molinos de marea de la Bahía de Cádiz (Ménanteau and Pourinet, 2002).

El ritual de elección de un nuevo obispo conllevaba a que las autoridades de la localidad vecina, San Fernando, trasladaran hasta el molino al nuevo obispo de la ciudad de Cádiz. Allí era recibido por las autoridades de esta segunda localidad. Tras la recepción en el Salón del Obispo, a las doce de la mañana las autoridades emprendían el regreso a la capital gaditana, acompañados del nuevo obispo.

### A.2.3.1. Ubicación

El molino se encuentra en el margen derecho de la carretera Nacional IV, que une Cádiz con San Fernando, haciendo límite entre ambas localidades. Está construido de forma perpendicular al caño del Río Arillo, limitando al norte con la Bahía de Cádiz y con la carretera anteriormente mencionada, la cual servía para el transporte de mercancías. Por tanto, el molino se encuentra situado dentro del término municipal de Cádiz.

Cabe destacar que el municipio de Cádiz se estructura en dos núcleos, co-

mo puede apreciarse en la Figura A.9: el núcleo urbano, y un gran espacio libre periurbano conectado de forma umbilical con la ciudad (Machuca Macías, 2013).



Figura A.9: Cádiz y el Parque Natural (Machuca Macías, 2013).

La zona del Parque Natural perteneciente al municipio de Cádiz está compuesta por tres salinas: las Salinas de San Félix, las Salinas de Nuestra Señora de los Dolores y las Salinas de Roquetas y Preciosas. El edificio objeto de estudio se encuentra situado en la primera de ellas (Véase Figura A.10), en una situación privilegiada, a ocho kilómetros del centro de Cádiz, y a dos del centro de San Fernando.

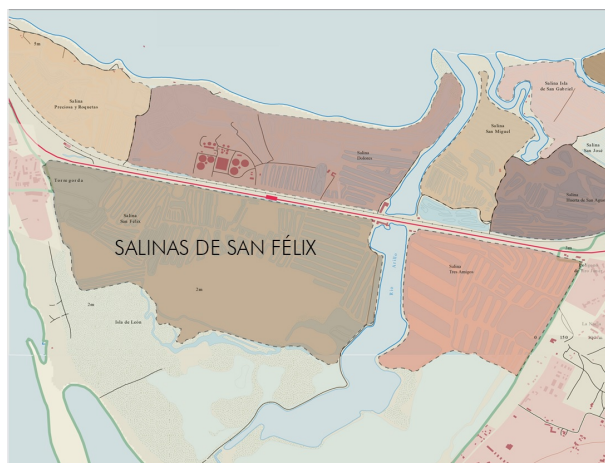


Figura A.10: Las Salinas de San Félix (Machuca Macías, 2013).

### A.2.3.2. Arquitectura del molino

Su construcción fue solicitada en marzo de 1798. La Junta de Fortificaciones de la plaza de Cádiz dio su visto bueno en el mismo mes. En su primera concepción, el molino estaba pensado para ser dotado de cuatro o seis piedras molidoras. Sin embargo, el ingeniero que proyectó y dirigió la obra, don Luis Huet, realizó una novedosa traza en forma de L, para aprovechar así el espacio que le permitía el caño, y dotándolo de doce piedras. Este molino, por tanto, fue una construcción novedosa en la bahía.

El edificio originario, de una planta de altura, consta de una larga nave de 46 metros de longitud por siete de ancho. A ambos lados de esta nave se encuentran anexados sendos recintos cuadrados, de la misma anchura que la nave, y de dos plantas de altura. Perpendicular al extremo izquierdo de la nave principal existe otro cuerpo de dos plantas de altura, de quince metros de longitud por ocho de anchura. En su cara este existen cuatro grandes arcos, por donde penetra el agua al subir la marea, hasta elevar las compuertas basculantes para llenar la presa. En la nave central y en su fachada norte existían dieciséis arcos, algunas de ellas derruidos como consecuencia de construcciones posteriores (Véase Figura A.11).

La piedra ostionera está presente tanto en la arcada inferior, debido a su resistencia al agua y al soporte que proporciona, como en el resto del edificio.

Posteriormente se han añadido pequeñas naves al edificio originario, algunas de ellas construidas parcialmente en ladrillo. En la Figura A.12 pueden



Figura A.11: Imagen de hipótesis de la forma en L primitiva del molino (Trocadero, 2011).

distinguirse estas nuevas construcciones, son la dependencia del molinero y el almacén.

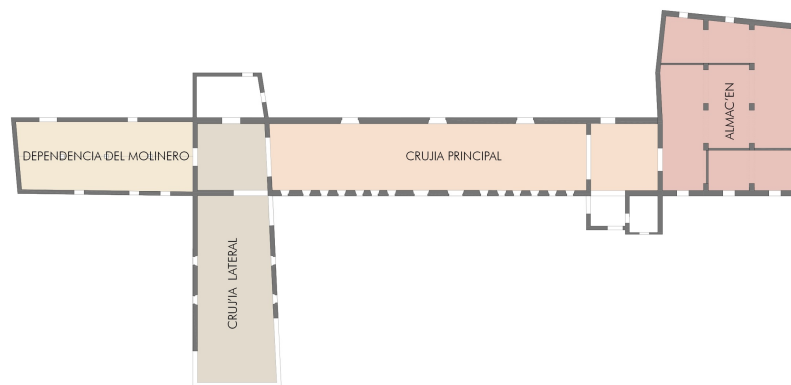


Figura A.12: Distribución interna del molino (Machuca Macías, 2013).

#### A.2.3.3. Estado actual del molino

En el verano de 1998, cumpliéndose dos siglos de su construcción, el molino sufrió un duro golpe consecuencia del paso del tiempo y del abandono de las autoridades, desplomándose uno de sus muros y toda la planta superior de una de sus alas. En las Figuras A.13 y A.14 se representa el estado actual del edificio.



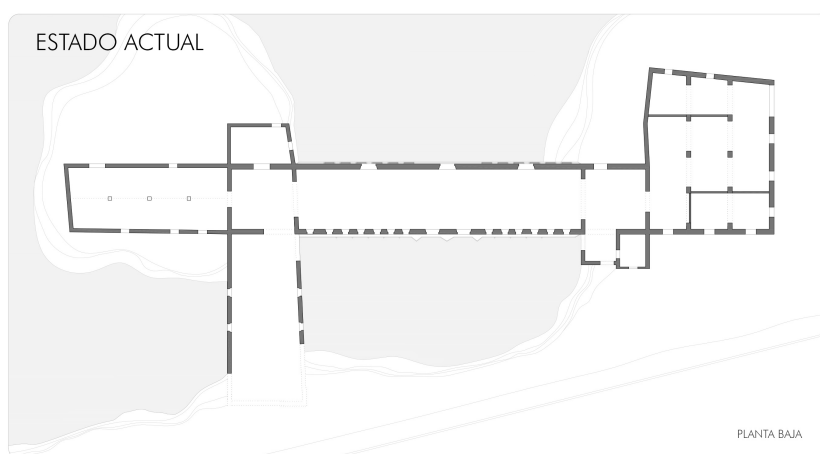


Figura A.13: Estado actual. Planta baja (Machuca Macías, 2013).

Actualmente el edificio se encuentra en una fase muy avanzada de daño, amenazando con ruina. Una actuación urgente es totalmente necesaria. Se espera que en los próximos años las autoridades competentes lleven a cabo trabajos de rehabilitación y adecuación del edificio, siguiendo el ejemplo de molinos vecinos que han sido rehabilitados y adecuados a otros usos. No se puede perder de vista que éste fue el molino más importante de los que funcionaron en la Bahía de Cádiz, por lo que la actuación sobre él está más que justificada.

En los últimos años se han llevado a cabo algunas intervenciones encaminadas a paliar la ruina inminente del molino, como el arriostramiento de muros a través de una estructura auxiliar de perfiles metálicos, como puede verse en las Figuras A.15, A.16 y A.17.

Como se indica en el trabajo realizado por Carmen Machuca, los daños que sufre este edificio se pueden clasificar de la siguiente forma:

- Pérdida de junta y redondeamiento de sillares o pérdida de sección.
- Pérdida de revoco.
- Fisuras, grietas y desprendimiento de paramentos.
- Drenaje deficiente.
- Vegetación enraizada.

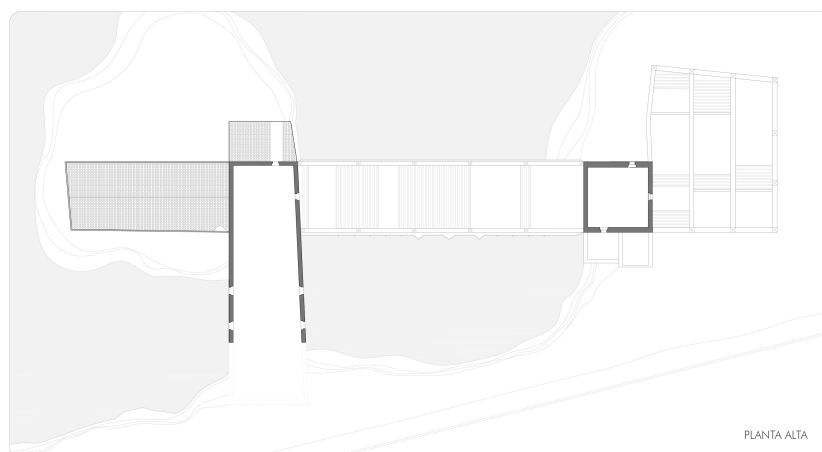


Figura A.14: Estado actual. Planta cubierta (Machuca Macías, 2013).

La preocupación por el estado en el que se encuentra este edificio es creciente entre los vecinos de la zona, pues no pasa desapercibido cuando se viaja entre Cádiz y San Fernando, convirtiéndose en un símbolo de abandono del valioso patrimonio de la Bahía de Cádiz (Véase Figura A.18).

El molino de marea es un buen exponente de lo que fue la tecnología tradicional de aprovechamiento de los recursos naturales, en este caso de los flujos y reflujos del mar.



Figura A.15: Estado actual. Interior. Elaboración Propia.



Figura A.16: Estado actual. Interior. Elaboración Propia.



Figura A.17: Estado actual. Exterior. Elaboración Propia.

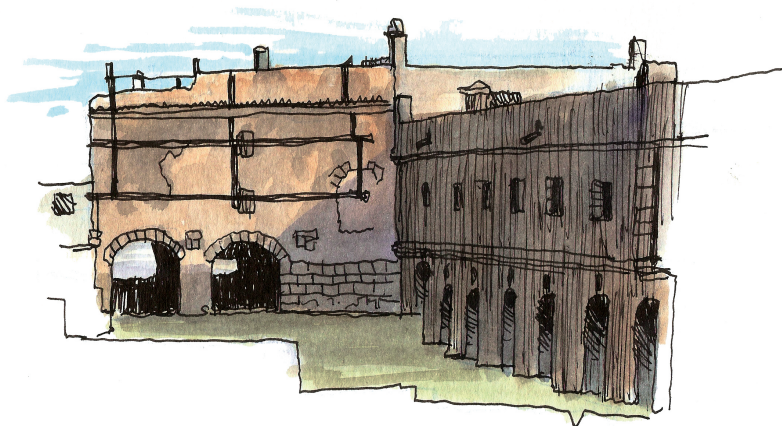


Figura A.18: Dibujo del molino hecho *in situ* con plumilla rellena de tinta china negra y con acuarela *Windsor and Newton* (Alcina Segura, 2016, p.199).

### A.3. Ejecución del levantamiento arquitectónico

#### A.3.1. Trabajos previos

Una vez justificada la elección del edificio sobre el que se ha realizado el levantamiento arquitectónico, se ha realizado una visita al edificio y a su entorno, con el fin de conocer la accesibilidad que presenta.

Para acceder al edificio es preciso desplazarse por la carretera Nacional IV o bien por el Sendero Tres Amigos, a pie. El edificio está rodeado de zonas fangosas, y parte de él está situado sobre el propio caño del Río Arillo que le da nombre. Además un lateral del edificio se encuentra paralelo a la carretera. Teniendo estos factores en cuenta, se aprecia que las zonas sobre las que es posible realizar el estacionamiento son limitadas.

Tras esta primera visita se determina la fecha en la que realizar el trabajo de campo de toma de datos. Para ello se ha escogido un día concreto teniendo en cuenta tanto la hora de la bajamar, como el coeficiente de marea. Éste debe ser alto para que la marea baje lo suficiente como para poder pisar las zonas fangosas, en caso de que fuera necesario.

Por otro lado, se ha realizado la planificación del trabajo de levantamiento, a través del modelo creado con *Net-Logo*.

Como se ha comentado anteriormente, el modelo de *Net-Logo* está en proceso de mejora continua. En el momento en el que se ha realizado este trabajo, el modelo ofrece la posibilidad de importar polígonos procedentes de otros programas, y escalar dicho polígono para que se adapte al tamaño del *Mundo* de *Net-Logo*. Sin embargo el modelo aún no es capaz de reconocer aquellas zonas sobre las que no es posible estacionar, debido a la presencia de agua, tráfico o zonas no accesibles. Por ello la solución obtenida es aproximada. Los pasos seguidos han sido los siguientes:

1. Digitalizar a través del *software QGIS* la planta del edificio, utilizando para ello la ortofoto *PNOA\_MA\_OF\_ETRS89\_HU29\_H50\_1068* descargada a través del Centro Nacional de Información Geográfica, (Instituto Geográfico Nacional, 2017) (Véase Figura A.19).
2. Exportar los datos vectoriales con la información sobre los vértices que definen el polígono del recinto en planta, a un fichero .CSV. Este fichero contiene las coordenadas de los puntos ordenadas según el orden de unión entre los mismos.

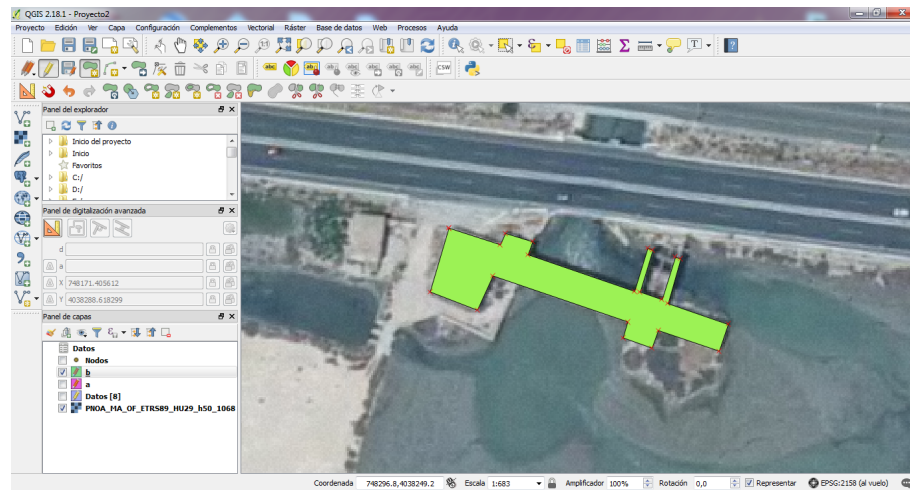


Figura A.19: Creación de capa vectorial en QGIS para la digitalización de la planta del edificio sobre ortofoto de la zona. Elaboración Propia.

3. Importar a *Net-Logo* el fichero .CSV con la planta del edificio. En este paso el polígono importado se escala para que se ajuste a la pantalla del *Mundo* de *Net-Logo*.
4. Fijar el rango de trabajo del instrumento (parámetros  $d$  y  $D$ ), así como el ángulo máximo admisible,  $\gamma_{maxadm}$ .
5. Inspeccionar el terreno.
6. Inicializar el algoritmo genético.
7. Obtener la solución óptima (Véase Figura A.20).

### A.3.2. Toma de Datos

Para la adquisición de datos se han utilizado los siguientes instrumentos:

- Escáner Láser Leica C10, con tecnología basada en tiempo de vuelo.
- Cámara panorámica iSTAR 360° de Nctech.
- Trípodes.
- Dianas reflectantes.

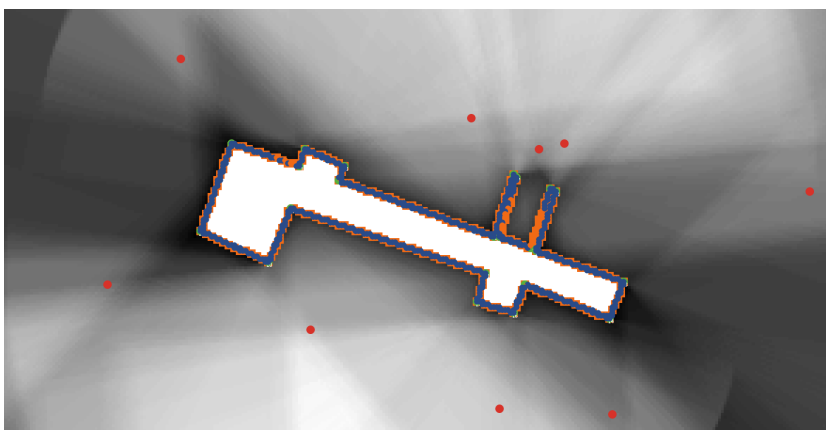


Figura A.20: Puntos de vista obtenidos tras la aplicación del AG. Elaboración Propia.

El campo de visión tanto del escáner como de la cámara panorámica es de  $360 \times 270^\circ$ , coincidiendo ambos. Por lo que, desde la misma posición en la que se ha realizado un escaneo, se ha tomado una fotografía panorámica.

La cámara utilizada está compuesta por cuatro objetivos gran angulares, como puede apreciarse en la Figura A.21. Con esta cámara se obtienen imágenes de alta resolución HDR.



Figura A.21: Cámara panorámica iSTAR 360° de Ntech. Elaboración Propia.

Cabe destacar que estos instrumentos son la última tecnología que puede utilizarse en este tipo de trabajo.

La toma de datos se ha realizado desde puntos próximos a los obtenidos a través del método diseñado, teniendo en cuenta las limitaciones que presentaba el entorno. Se han realizado un total de nueve escaneos, a una resolución



de 1 centímetro a una distancia de 10 metros.

### A.3.3. Post-procesado y resultados obtenidos

Los datos obtenidos en los nueve escaneos han sido registrados a través del *software Cyclone* de Leica. El registro de las nubes de puntos se ha realizado por superposición. En algunas ocasiones este proceso ha sido laborioso debido a la falta de solape entre las distintas nubes de puntos. En la Figura A.22 pueden apreciarse las nueve nubes de puntos coloreadas y unidas, así como las posiciones desde donde se han tomado los datos.

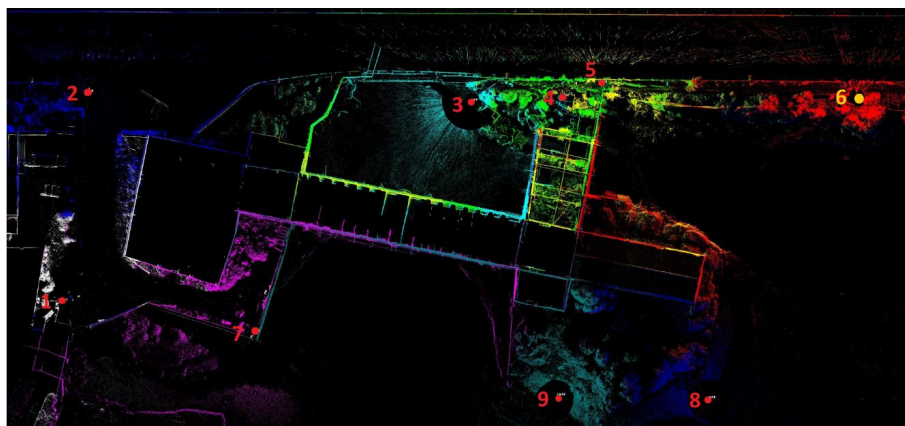


Figura A.22: Nubes de puntos registradas y posición de cada uno de los escaneos. Elaboración Propia.

Por otro lado, la textura de los puntos se ha añadido a través del *software ColourCloud*, de Ntech. El programa es capaz de reconocer automáticamente los puntos a los que debe asociar la textura correspondiente. Cabe mencionar la calidad de las imágenes aplicadas a la textura. Los escáneres láser suelen incorporar su propia cámara de fotos para colorear las nubes de puntos, pero la calidad de éstas son mucho más baja que la obtenida con la cámara utilizada, quedando un resultado bastante pobre.

En las Figuras A.23 y A.24 pueden apreciarse los alzados principales del edificio. Por otro lado, en la Figura A.25 puede verse una vista en perspectiva de la nube de puntos completa y texturizada del edificio.

Para concluir, esta nube de puntos se ha georreferenciado en ETRS89 usando la ortoimagen *PNOA\_MA\_OF\_ETRS89\_HU29\_H50\_1068* del Centro Nacional de Información Geográfica. Puede apreciarse en la Figura A.26.



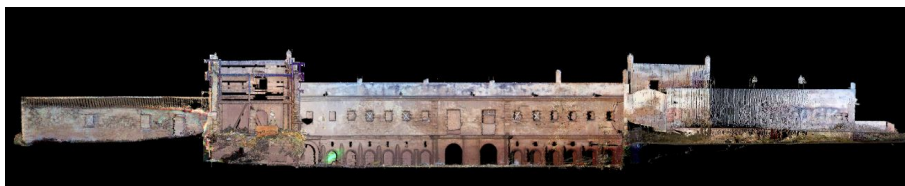


Figura A.23: Alzado norte visualizado desde *Cyclone*. Elaboración Propia.

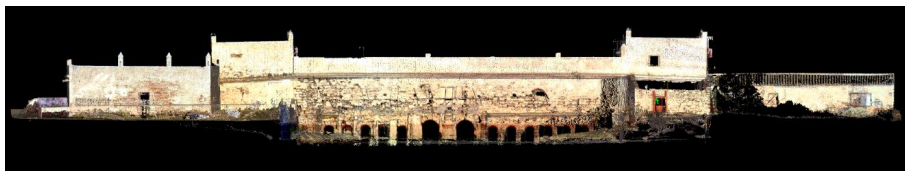


Figura A.24: Alzado sur visualizado desde *Cyclone*. Elaboración Propia.

#### A.3.4. Conclusión

Se concluye que el modelo diseñado en *Net-Logo* desarrollado en la Sección 5.3 ha sido de utilidad para la aplicación sobre un edificio concreto, en aras de ejecutar el levantamiento arquitectónico del mismo.

Cabe destacar que el modelo presentado permite la elección de los puntos de vista de forma manual. En este caso, al existir zonas sobre las que no ha sido posible estacionar, la opción de elección de puntos de forma manual resulta muy interesante, ya que puede que el AG localice componentes sobre zonas donde no es posible estacionar. Además, esta opción abre la posibilidad de comprobar *a posteriori* de la realización de un trabajo, si los puntos elegidos han sido suficientes para tomar de forma correcta los datos geométricos.

No obstante, se está trabajando para que el propio modelo tenga en cuenta aquellas zonas sobre las que no es posible posicionarse, para que el AG no coloque a ningún componente sobre estas zonas, y así reducir el campo de búsqueda a la situación real.

Por último, resaltar que los datos geométricos obtenidos constituyen una importante fuente de información geométrica a tener en cuenta en trabajos de rehabilitación de este edificio, y que se espera poder aportar para contribuir a su intervención.

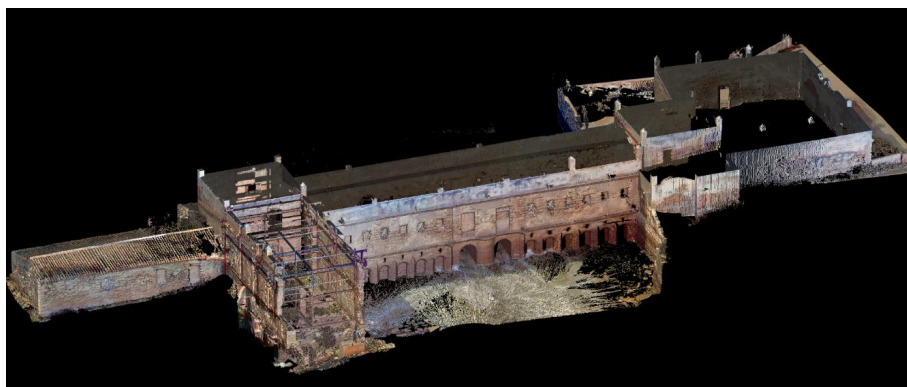


Figura A.25: Vista en perspectiva. Elaboración Propia.



Figura A.26: Georreferenciación de la nube de puntos a través del método Helmert. Elaboración Propia.

## Apéndice B

### Estudio de ángulos

A continuación se describen los trabajos que se han realizado para estudiar y analizar la precisión que se obtiene al tomar datos sobre un paramento, en función al ángulo de incidencia sobre el mismo. La toma de datos se ha realizado utilizando un escáner láser 3D de la casa Leica, modelo C10. El ángulo objeto de estudio al que a partir de ahora se hará referencia, es el formado entre la normal del paramento en el punto de lectura, y el rayo de incidencia del escáner sobre el mismo. A este ángulo se le va a llamar en toda esta sección  $\gamma$ , y es el que aparece reflejado en la Figura B.1, en la cual puede apreciarse como el valor del mismo aumenta conforme aumenta la distancia desde el escáner al punto leído.

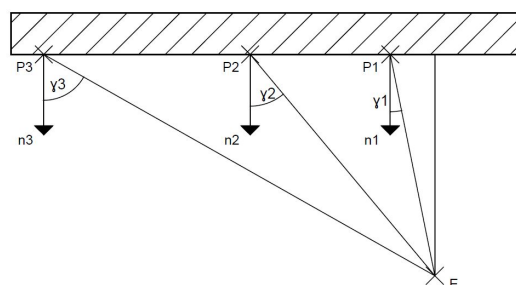


Figura B.1: Variación del ángulo  $\gamma$ . Elaboración Propia.

En esta sección se desarrolla el estudio llevado a cabo para conocer la influencia del ángulo  $\gamma$  en la correcta toma de datos.

Gustavo Olague (Olague et al., 2000), al definir las variables que influyen en el Algoritmo Genético que diseña (Apartado 1.5.3.4), indica que una de

ellas es el ángulo de incidencia. El ángulo máximo de incidencia admisible depende del tipo de instrumento, de la geometría y el material del paramento, de la distancia desde la cual se realiza la lectura, entre otros factores.

En una investigación llevada a cabo por el Equipo Investigador en Geomática, Topografía y Cartografía (GTC) de la Universidad de Oviedo (García et al., 2013) se realizan pruebas con distintos materiales de uso frecuente en la construcción: cemento, granito, ladrillo, madera, etc. Los resultados obtenidos pueden verse en la Figura B.2. Este estudio se realizó con una estación total computarizada Leica TPS 700, y las conclusiones obtenidas son las siguientes:

- La distancia máxima de medición debe ser inferior a 50 m.
- El ángulo de incidencia máximo debe ser inferior a  $45^\circ$

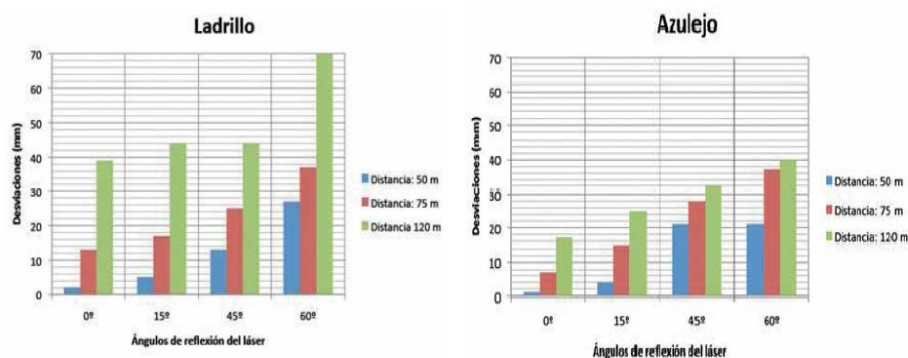


Figura B.2: Resultados obtenidos en el análisis de precisiones del distanciómetro láser sobre diferentes materiales. Gráficos de respuesta del azulejo y del ladrillo con estación total computarizada Leica TPS 700 (García et al., 2013).

Debido a que el modelo diseñado en el presente trabajo de investigación tiene en cuenta el valor del ángulo máximo admisible que puede tomar  $\gamma$ , como puede verse en los Apartados 5.2.4 y 5.3.3.1, se ha procedido a realizar pruebas en distintos paramentos. Cabe destacar que, como se ha mencionado anteriormente, la influencia de este ángulo puede variar dependiendo de muchos factores, por lo que una investigación a fondo sería necesaria en este tema concreto. No es el objetivo de este trabajo hacer una investigación exhaustiva centrada en el ángulo de incidencia entre el instrumento y el paramento leído, sino simplemente analizar una serie determinada de datos que serán decisivos en los estudios de caso que se han realizado para testear las aplicaciones desarrolladas.

Dicho esto, se procede a exponer de forma detallada el proceso que se ha llevado a cabo.

### B.1. Toma de datos

Se han escaneado diferentes paramentos. En cada uno de ellos se han colocado cuatro dianas: una en la posición en la cual el valor de  $\gamma$  es igual a  $0^\circ$ , y otras tres en las cuales el valor de  $\gamma$  es de  $45^\circ$ ,  $60^\circ$  y  $70^\circ$ , respectivamente.

En primer lugar se ha realizado un escaneo a baja resolución, y posteriormente se ha realizado un segundo escaneo por cada paramento a alta resolución sobre las dianas. El escáner utilizado es capaz de reconocer las dianas que se han usado, realizando una toma de datos de gran precisión sobre ellas, obteniendo así la coordenada del centro de la diana escaneada.

- Primera toma de datos: Paramento de Mármol (Paramento M).

El escáner láser se ha colocado a una distancia de 2,32 metros frente a la pared de mármol, perteneciente a un habitáculo interior.

Se han colocado las distintas dianas por cada valor de  $\gamma$ . Estas dianas se han etiquetado en función al material y al ángulo, recibiendo los siguientes nombres: 0M para la diana colocada a  $0^\circ$  sobre el aplacado de mármol, 45M, 60M y 70M, respectivamente.

A continuación, se ha realizado un escaneo a una resolución baja (2 cm a 5 metros). Posteriormente se ha realizado un escaneo de alta precisión sobre las cuatro dianas. Este escaneo ha servido para hallar las coordenadas de los centros de las dianas de una forma más precisa.

- Segunda toma de datos: Paramento de Fábrica de Ladrillo Visto (Paramento FLV).

El segundo escaneo se ha realizado sobre una pared de fábrica de ladrillo visto situada en el exterior. El escáner se ha colocado a una distancia de 3,37 metros con respecto al paramento. Se ha calculado la posición de las dianas para que el valor de  $\gamma$  sea 0, 45, 60 y 70 grados. Las dianas han sido etiquetadas como 0FLV, 45FLV, 60FLV y 70FLV.

A continuación, se ha realizado el escaneo a una resolución baja de 2,5 cm a 5 metros, y posteriormente se ha escaneado con más exactitud las dianas.

- Tercera toma de datos: Paramento de Hormigón Visto (Paramento HV).

Siguiendo el mismo procedimiento que el usado para los dos paramentos anteriores, se ha escaneado esta vez un muro de hormigón visto. La distancia a la que se ha colocado el escáner con respecto al paramento ha sido de 3,73 metros. Se han colocado las dianas en las posiciones deseadas y se ha escaneado con resolución 2,5 cm a 5 metros. Las dianas han sido etiquetadas como 0HV, 45HV, 60HV y 70HV.

Algunas de las fotos del proceso de toma de datos pueden verse reflejadas en las Figuras B.3, B.4,



Figura B.3: Lectura de paramento de mármol y dianas colocadas. Elaboración Propia.

## B.2. Manejo de nubes de puntos

Tras haber obtenido las distintas nubes de puntos, cada una de ellas es nombrada de forma que puedan ser identificadas (M, FLV o HV). Por cada escaneo se ha obtenido una nube de puntos dispersa, y otra con más resolución alrededor de las dianas, siendo capaz así de conocer con exactitud las coordenadas de cada una de las dianas. Los datos de los que se parte son los siguientes:



Figura B.4: Lectura de paramento de fábrica de ladrillo visto con escáner láser. Elaboración Propia.

- Escaneo\_M.PTS: es la nube de puntos obtenida del primer escaneo, en la cual se leyó el paramento de mármol.
- Dianas\_M.PTS: son las coordenadas X,Y,Z de las cuatro dianas que se han leído en el primer escaneo.
- Escaneo\_FLV.PTS: se refiere a la nube de puntos obtenida tras el escaneo a la superficie de fábrica de ladrillo visto.
- Dianas\_FLV.PTS: son las coordenadas X,Y,Z de las cuatro dianas colocadas sobre el paramento de ladrillo.
- Escaneo\_HV.PTS: es la nube de puntos obtenida tras el tercer y último escaneo sobre la superficie de hormigón visto.
- Dianas\_HV.PTS: archivo con las coordenadas X,Y,Z de las cuatro dianas colocadas sobre el paramento de hormigón visto.

El procedimiento que a continuación se detalla ha servido para obtener la precisión con la que se puede obtener la coordenada de un punto cualquiera  $P_i(X_i, Y_i, Z_i)$ . Para ello se ha comparado el valor de las coordenadas de aquellos puntos en los que se ha colocado una diana (y, por tanto, cuyas coordenadas se conocen con precisión), con el valor obtenido al aproximar una serie de puntos dispersos leídos alrededor de cada una de las dianas, a baja resolución. Se ha limitado por tanto un área alrededor de cada una de las dianas equivalente al de una circunferencia de 15 centímetros de radio, y se ha aproximado por mínimos cuadrados esos puntos a una superficie. Tras esto, se ha comparado la coordenada del centro de la superficie creada con la coordenada de la diana correspondiente, hallándose así una forma de medir el error existente.

Para cada uno de los escaneos se repite el siguiente procedimiento, siendo X igual a M, FLV o HV, en función del paramento del que se trate.

- Se importa el archivo de nube de puntos Escaneo\_X.PTS al *software Autodesk RECAP*, creándose el archivo Escaneo\_X.RCP.
- Este archivo Escaneo\_X.RCP es importado al *software Autodesk AutoCAD*, creándose el archivo Escaneo\_X.DWG.
- Sobre el archivo Escaneo\_X.DWG son introducidos los siguientes elementos:
  - Puntos con las coordenadas de los archivos Dianas\_X.PTS.
  - Rayo de visión desde la posición del escáner, es decir, el punto de coordenadas 0,0,0 del sistema, a cada uno de los puntos dibujados en el paso anterior, es decir, los puntos cuyas coordenadas coinciden con las coordenadas de las cuatro dianas correspondientes.
  - Con centro en cada una de las dianas es dibujada una circunferencia de 15 centímetros de radio. Las circunferencias se trazan contenidas en un plano definido por tres puntos del paramento.
  - Aquellos puntos de la nube de puntos que han quedado dentro de esa circunferencia, son copiados a una nueva capa. Obsérvese que cuanto mayor es el ángulo  $\gamma$ , menor es el número de puntos que queda alrededor de las dianas, pues los puntos se encuentran cada vez más dispersos.
  - Los puntos obtenidos en el paso anterior se exportan a un fichero denominado PuntosCercanosEscaneo\_X.TXT.
  - Los puntos del fichero PuntosCercanosEscaneo\_X.TXT se importan en el *software Geomagic*, obteniéndose así el archivo PuntosCercanosEscaneo\_X.WRP.
  - Con la herramienta «Rasgos→ Plano→ Mejor Adaptación» se realiza la aproximación por mínimos cuadrados de cada conjunto de puntos cercanos a cada una de las dianas. De esta forma se consigue la superficie que mejor se ajusta a los puntos que se encuentran más cerca de cada una de las dianas.
  - Por cada diana se genera un plano, cada uno de los cuales son exportados en formato .SAT nombrándolos de la siguiente forma: Escaneo\_X\_ $\gamma$ .SAT, pudiendo variar  $\gamma$  entre 0, 45, 60 o 70, en función a la diana de la que se trate.



- Sobre el archivo Escaneo\_X.DWG se importan cada uno de los planos Escaneo\_X\_γ.SAT, los cuales se importan como superficies.
- Se calcula el centro de cada una de las superficies importadas Escaneo\_X\_γ.SAT. Esos puntos son nombrados como Escaneo\_X\_γprox.
- Las coordenadas de las dianas son comparadas con la correspondiente coordenada Escaneo\_X\_γprox para calcular el error.

En la Figura B.5 puede verse la nube de puntos dispersa obtenida al leer el paramento FLV, así como los rayos de visión partiendo de la posición del escáner, a cada una de las coordenadas de las dianas. También puede apreciarse las superficies creadas con los puntos cercanos a la posición de las dianas.



Figura B.5: Nube de puntos dispersa del paramento FLV. Rayos lanzados desde el escáner a cada una de las dianas, así como superficies planas creadas alrededor de cada una de las dianas. Elaboración Propia.

### B.3. Error calculado y análisis de los datos

Tras haber realizado el procedimiento descrito anteriormente para cada uno de los escaneos, se han obtenido los siguientes gráficos.

El gráfico reflejado en la Figura B.6 da muestra de cómo la densidad de puntos disminuye conforme el ángulo  $\gamma$  aumenta. La densidad de puntos existentes alrededor de las dianas colocadas varía en función al material, y disminuye conforme aumenta el ángulo de incidencia. Esto significa que también influye la reflectividad del material leído.

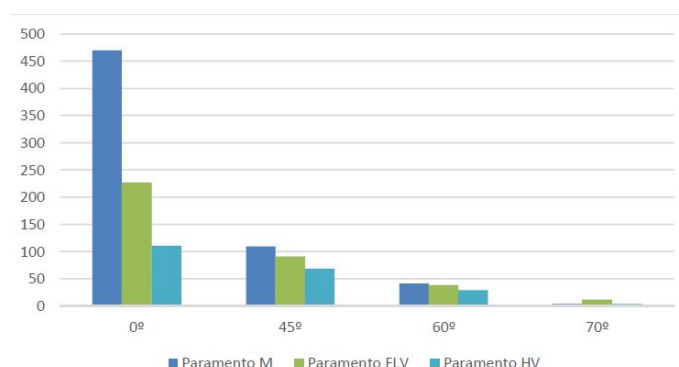


Figura B.6: Análisis de la variación de la densidad de puntos alrededor de cuatro puntos ubicados en ángulos determinados (0°, 45°, 60°, 70°), y en tres tipos de paramentos distintos (M, FLV y HV). Elaboración Propia.

Por otro lado, en el gráfico mostrado en la Figura B.7 se aprecia el error expresado en milímetros que se está asumiendo al tomar puntos sobre los distintos paramentos, según el ángulo de incidencia  $\gamma$ . Como puede apreciarse, conforme el valor de  $\gamma$  aumenta, el error también aumenta, siendo éste último inversamente proporcional a la densidad de puntos, como no podía ser de otra forma. Aunque en los paramentos M y FLV el error es controlado cuando el valor de  $\gamma$  es de 70°, el paramento HV no presenta el mismo comportamiento. De ello se concluye que a partir de un valor de  $\gamma$  mayor a 60°, no es recomendable realizar una toma de datos, puesto que los datos leídos pueden ser de baja calidad.

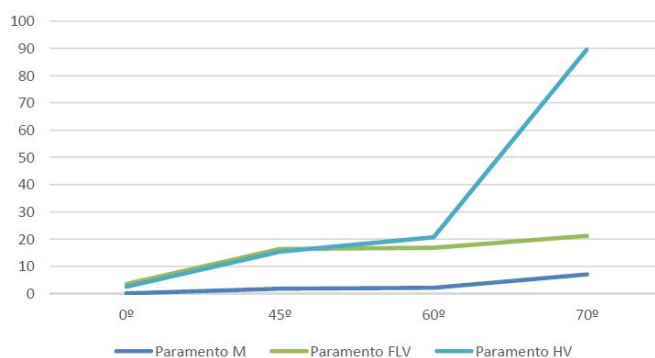


Figura B.7: Variación del error en tres tipos de paramentos distintos (M, FLV y HV), en función al valor del  $\gamma$ . Elaboración Propia.

## Apéndice C

# Protocolo ODD

Como se ha mencionado en el Apartado 5.3.3.1, con el fin de implementar el AG de forma satisfactoria en *Net-Logo*, como un sistema multiagente, es necesario desarrollar el protocolo conocido como **Protocolo ODD** -*Overview, Design Concepts and Details* (Grimm et al., 2010), cuya función es la de ordenar y estructurar los problemas y los agentes que participan en el algoritmo. En el apartado mencionado anteriormente, se describieron las partes más importantes de este protocolo. Sin embargo, en este apéndice se describe de forma completa. Aunque el protocolo original tiene más apartados, no todos tienen porqué ser aplicables a todos los modelos basados en agentes.

### 1. PROPÓSITO

- ¿Cuál es el propósito del modelo?  
Obtener el número mínimo de Observadores, así como la posición de los mismos, desde los cuales visualizar las fachadas exteriores de un edificio, con el fin de realizar un levantamiento arquitectónico.
- ¿Con qué objetivo se ha desarrollado?  
El objetivo con el que se desarrolla el modelo, es el de minimizar el tiempo de trabajo, así como asegurar que la toma de datos al realizar el levantamiento arquitectónico es completa, y no se quedan zonas sin visualizar.
- ¿Para qué se va a utilizar?  
Se va a utilizar para obtener un método a través del cual, cualquier persona que desee realizar un levantamiento arquitectónico, lo ha-

ga con la seguridad de que el resultado será bueno, y el tiempo de trabajo será mínimo.

## 2. ENTIDADES, VARIABLES DE ESTADO Y ESCALAS

- ¿Qué tipo de entidades conforman el modelo (agentes/individuos, unidades espaciales, medio ambiente, colectividades)?
  - Los agentes tipo tortugas:
    - Observadores.
    - Recorredores.
    - Vértices que definen la planta del edificio.
  - Los agentes tipo link:
    - Fachadas del edificio.
  - Medio ambiente:
    - Interior del edificio.
    - Exterior del edificio.
    - Borde del edificio.
  - Colectividades:
    - Individuo del algoritmo genético: compuesto por tortugas del tipo *Observadores*.
    - Planta del edificio: formada por tortugas tipo *vértices* y por agentes link tipo *fachadas*.
- ¿Qué variables de estado o atributos internos caracterizan a tales entidades?
  - Observadores: Tienen un alcance mínimo y máximo de trabajo, y un ángulo de apertura de visión que puede ser limitado, o llegar hasta 360°. Los observadores pueden estar en estado apagado o encendido.
  - Recorredores: son elementos de la fachada del edificio que van dejando un rastro que marca desde dónde es posible vigilar ese punto. El rastro que deja el recorredor está comprendido entre una distancia mínima y una distancia máxima, dentro de la cual se puede observar bien al recorredor. Tienen una apertura limitada, en función al ángulo mínimo de incidencia. El rastro se dibuja hacia el lado exterior del edificio y con su eje de simetría perpendicular al paramento desde donde es lanzado.
  - Vértices del edificio: Limitan las fachadas del edificio. Se definen dibujándolos con el ratón en la posición en la que interese. Es interesante conocer sus coordenadas. Se introducen

---

de forma ordenada, siendo adyacentes aquellos que se dibujen consecutivamente. También existe la opción de generarlo de forma aleatoria. Se está trabajando en la importación de estos vértices a través de archivos CSV con información generada en *software* externo.

- Fachadas del edificio: dan forma a la planta del edificio, y tienen que ser completamente vigiladas. Separan el interior del exterior del edificio. Quedan limitadas entre dos vértices consecutivos.
  - Interior del edificio: está coloreado de forma distinta al exterior y al contorno, de color blanco. En esta área no se puede ubicar ningún observador.
  - Exterior del edificio: no está coloreado. En este área se pueden ubicar los observadores.
  - Contorno del edificio: es el límite entre el interior y el exterior, y es lo que debe ser visualizado.
  - Individuo del algoritmo genético: se expresa con las coordenadas cartesianas de los Observadores que lo componen.
- ¿En qué unidades se expresan tales variables o atributos?
    - Observadores: en coordenadas cartesianas.
    - Recorredores: se deslizan sobre las fachadas.
    - Vértices del edificio: en coordenadas cartesianas.
    - Fachadas del edificio: entre dos vértices que vendrán dadas por sus coordenadas cartesianas.
    - Individuo del algoritmo genético: se expresa con las coordenadas cartesianas de los Observadores que lo componen, y con el número de observadores que lo componen, así como su estado, pudiendo variar entre activados y desactivados.
  - ¿Cuál es la extensión espacial y temporal del modelo?
    - La extensión espacial será la planta de un edificio a una escala proporcionada.
    - La extensión temporal será por tiempo de ejecución.
  - ¿Con qué nivel de precisión espacial y temporal se realizará la simulación?
    - Precisión espacial: alto.
    - Precisión temporal: bajo.

### 3. RESUMEN DEL PROCESO Y SU PLANIFICACIÓN

- ¿Qué entidad hace qué?
  - Observadores: vigilan las fachadas de un edificio desde el exterior del mismo, con un alcance limitado.
  - Recorredores: lanzan un rastro de las ubicaciones desde las cuales ese punto es visualizado correctamente.
  - Vértices que definen la planta del edificio: delimitan la forma del edificio a vigilar.
  - Fachadas del edificio: delimitan la forma en la que los vértices se unen, así como la separación entre el interior y el exterior del edificio.
  - Interior del edificio: supone un obstáculo en sí mismo para la visualización del edificio.
  - Exterior del edificio: es el área desde la cual se puede observar al edificio.
  - Individuo del algoritmo genético: el conjunto de los observadores que componen un individuo, es capaz de vigilar por completo el edificio.
- ¿En qué orden se ejecutan los diferentes procesos?
  - Se sitúan los vértices que conforman la planta del edificio, en el orden adecuado.
  - Se colorea el borde del grafo que representa la planta del edificio.
  - Se colorea de otro color el interior del grafo que representa la planta del edificio.
  - Se lanza una agente recorredor entre vértice y vértice, que desde cada *patch* perteneciente a la fachada emite un cono de visión hacia el semiplano exterior del polígono que simboliza el área desde la cual ese punto es vigilado.
  - Se contabiliza la cantidad de veces que un *patch* del exterior del edificio ha pertenecido a algún cono.
  - El color del *patch* se intensifica cuanto mayor sea el número de conos de visión que lo hayan abarcado.
  - Se inicializa el Algoritmo Genético:
    - Se genera la población inicial con un número determinado de individuos. Cada individuo tiene el mismo número de observadores. Se genera de forma aleatoria las posiciones de estos observadores, y se elige, de forma aleatoria, cuáles de ellos están activados y cuáles desactivados.

- 
- Se evalúan los individuos a partir de la Función Objetivo, dependiendo de: la cantidad de fachadas que el individuo es capaz de vigilar; el número de observadores; la visibilidad directa entre un observador y otro; y la calidad con la que una fachada está siendo visualizada.
  - Se selecciona un porcentaje de los individuos que van a poder generar descendencia. La selección se hace por torneo.
  - Se produce el cruce entre dos individuos seleccionados de forma aleatoria. Se elige aleatoriamente el punto de corte a partir del cual se van a mezclar los genes de los padres.
  - Se aplica la mutación, con una probabilidad determinada. La mutación consiste en modificar de forma aleatoria las coordenadas de un observador.

#### 4. CONCEPTOS DE DISEÑO

##### ■ Principios fundamentales:

¿Qué conceptos, teorías, hipótesis teóricas subyacen en el diseño del modelo?

Hay que tener en cuenta el funcionamiento de los aparatos de toma de datos, los cuales tienen un alcance mínimo y máximo de trabajo, así como un ángulo de apertura determinado.

También, hay que considerar que una fachada no se vigila correctamente si no se hace con un ángulo de incidencia aproximadamente mayor a 20°.

Por otro lado, se sabe que para conectar diferente tomas de datos de diferentes observadores, lo más sencillo sería que existiera visibilidad directa entre ambas. Pero en la realidad, se sabe que llegado el momento se podrá solucionar de alguna otra forma, en el caso de que dos observadores sean capaces de visualizar una zona común.

##### ■ Objetivos:

- ¿Qué objetivos persiguen los individuos mediante los procesos de adaptación que rigen sus comportamientos?
  - Observadores: vigilar una fachada.
  - Recorredores: lanzar un cono de visión y desplazarse a lo largo de una fachada.
  - Vértices que definen la planta del edificio: definir la dirección en la que se tienen que desplazar los recorredores.

- Fachadas del edificio: separar interior de exterior y dar información para colorear el interior del edificio, así como servir de obstáculo ante las áreas de visibilidad lanzada.
- Interior del edificio: no permitir que se coloquen observadores en su interior.
- Exterior del edificio: permitir que se coloquen observadores en su interior y almacenar la información de cuántas áreas de visibilidad han pasado por encima de cada uno de los *patches* que componen el exterior.
- Contorno del edificio: separar interior de exterior y dar información para colorear el interior del edificio.
- Individuo del algoritmo genético: visualizar todas las fachadas del edificio.
- Interacción:
  - ¿Qué tipos de interacciones se asumen como relevantes entre los agentes?  
Es relevante la interacción entre los agentes que representan a los Observadores, y los agentes que representan las fachadas, ya que desde los Observadores se deben vigilar todas las fachadas, y con una calidad adecuada.
  - ¿Se trata de interacciones directas, en las que los encuentros entre agentes influyen sobre los mismos?  
Sí, son interacciones físicas.
- Aleatoriedad: ¿Qué procesos se han modelado asumiendo que son, total o parcialmente, aleatorios?
  - La ubicación de los agentes que representan a un Observador, se genera de forma aleatoria.
  - También, el cruce entre dos individuos incluye el proceso de corte aleatorio.
- Observación:
  - ¿Qué datos se generan y recopilan a partir de la simulación a efectos de análisis?  
Se recopila información de cuántos Observadores son necesarios, en función a lo complejo que sea el edificio.
  - ¿Cómo son recopilados tales datos, y en qué momento o momentos?  
Se recopila cuando se paraliza el algoritmo.

## 5. INICIALIZACIÓN



- 
- ¿Cuál es el estado inicial del modelo, esto es, en el momento  $t=0$  de la ejecución de la simulación?

En el tiempo cero, lo que aparece en pantalla es negro, para que se introduzca la ubicación de los vértices que conforman la planta del edificio.

- ¿Cuántas entidades forman la sociedad virtual inicialmente, y qué valores, exactos o como distribución aleatoria, tienen las variables de estado de las entidades?

Lo primero que se va a introducir es la ubicación de los vértices del edificio, así como la unión entre los vértices. A continuación se genera la población inicial compuesta por puntos de vista distribuidos aleatoriamente alrededor del edificio.

## 6. DATOS DE ENTRADA

- ¿Utiliza el modelo datos de fuentes externas (ficheros de datos, u otros modelos) para representar procesos que varían en el tiempo durante la simulación?

La opción de importar una imagen, o un archivo compuesto por puntos, en el cual aparezcan las coordenadas de los vértices, así como el orden de unión entre ellas, es realmente útil. Esto permite trabajar con formas reales de edificios, y con escalas reales. En estos momentos se está trabajando en ello.



## Apéndice D

# Código del modelo en *Net-Logo*

```
;----- Variables -----  
extensions [csv]  
;Familias de agentes en el modelo.  
breed [vertices vertice]  
breed [recorredores recorredor]  
breed [fachadas fachada]  
breed [observadores observador]  
breed [individuos individuo]  
;Propiedades de los vértices (esquinas que definen la fachada del edificio).  
vertices-own[etiqueta]  
;Propiedades de los recorredores (agentes que recorren la fachada del edi-  
ficio).  
recorredores-own[  
  origen  
  angulo-incidencia  
  patch-inicio
```

patch-fin ]

;Propiedades de la fachada (agentes tipo link que se definen entre dos agentes tipo vértice).

fachadas-own[

orientacion-cono

]

;Propiedades de los observadores (agentes que ven desde fuera del edificio la fachada del mismo).

observadores-own[

numero-paramentos

angulo-minimo

]

;Propiedades de los individuos (agentes que representa una solución del problema, con un conjunto de posiciones correspondientes a los observadores y ;una lista que indican si están activados o no).

individuos-own [

;Genes lista-observadores lista-activaciones

;Propiedades fitness ]

;Propiedades del suelo en el modelo.

patches-own[

interior

vertice-encima

valor-observado

recorredorVisto

recorredores-vistos ]

;Variables globales del modelo.

globals[

---

```
pulsado
etiqueta-global
clock
poder-mover
grafo-finalizado
contador-recorredores
contador-recorredores-vistos
numero-observadores ;El numero de observadores determina la longitud
de los genes de cada individuo.
csv
point-list ]
;----- Limpiar mundo -----
;Método para limpiar el mundo.
to borrar-todo
_clear-all-and-reset-ticks
set contador-recorredores 0
set contador-recorredores-vistos 0
ask patches
[
set vertice-encima false
]
set grafo-finalizado false
reset-timer
end
;----- Planta del edificio -----
```

```

to crear-mapa
import-pcolors-rgb "mapa.png"
end

;----- Modo manual -----
;Método para dibujar los vértices con el ratón.
to dibuja-vertices
if mouse-inside? and mouse-down? != pulsado
[
ifelse grafo-finalizado = false
[
let m-xcor mouse-xcor
let m-ycor mouse-ycor
aerodinamico m-xcor m-ycor
][
user-message( "No se puede modificar el grafo una vez finalizado.")
]
]
set pulsado mouse-down?
end

to aerodinamico [punto-xcor punto-ycor]
ask patch punto-xcor punto-ycor
[
if vertice-encima = false
[
sprout-vertices 1

```

---

```
[
  set shape circle"
  set color green
  set label-color white
  set size 3
  set etiqueta who
  set etiqueta-global who
]
crear-arista-anterior
ver-etiquetas
set vertice-encima true
]
] end
```

;Método para crear aristas entre los distintos vértices pero siempre con el anterior.

```
to crear-arista-anterior
  if etiqueta-global != 0
  [
    ask vertices with[etiqueta = etiqueta-global]
    [
      create-link-with one-of other vertices with[etiqueta = etiqueta-global - 1]
    ]
  ]
end
```

;Método para mostrar las etiquetas de los nodos

```
to ver-etiquetas
ask vertices
[
set label who
set label-color white
]
end

;Método para calcular las etiquetas del vecino.
to-report calcular-etiqueta-vecino [etiq]
report (etiq + 1) mod (etiqueta-global + 1)
end

;Método para cerrar el grafo, conecta el último vértice con el cero.
to cerrar-grafo
ifelse(etiqueta-global <2)
[
user-message( "No puede existir un modelo con menos de tres vértices.")
]
[
ifelse grafo-finalizado = false
[
ask vertices[
if etiqueta = 0[
create-link-with one-of other vertices with[etiqueta = etiqueta-global]
]
]
]
]
```



---

```

]
[
user-message( "No se puede modificar el grafo una vez finalizado.")
]
]
end

;Método para poder mover el grafo por el entorno.
to mover-grafo
if mouse-down?
[
ifelse grafo-finalizado = false
[
let grabbed min-one-of vertices [distancexy mouse-xcor mouse-ycor]
while [mouse-down?]
[
ask grabbed [ setxy mouse-xcor mouse-ycor ]
display
]
][
user-message( "No se puede modificar el grafo una vez finalizado.")
]
]
end

;Método para terminar el grafo y dejarlo definitivo y coloreado.
to finalizar-grafo

```

```

ifelse(etiqueta-global <2)
[
  user-message( "No puede existir un modelo con menos de tres vértices.")
]
[
  ifelse grafo-finalizado = false
  [
    let angulo 0
    ask vertices
    [
      hatch 1
      [
        let etiqueta-vecino calcular-etiqueta-vecino etiqueta
        let vecino vertice etiqueta-vecino
        face vecino
        while [(distance vecino) >0.5]
        [
          ask patch xcor ycor
          [
            if (pcolor != orange) [set pcolor orange ask neighbors [set pcolor orange]]
            sprout-recorredores 1
            [
              setxy xcor ycor
              face vecino
              set contador-recorredores contador-recorredores + 1
            ]
          ]
        ]
      ]
    ]
  ]
]

```

---

```
set color orange
ask patch-right-and-ahead 90 1
[
if pcolor = black [set pcolor white]
]
rt -90
]
]
fd 0.5
]
die
]
]
colorea-interior
set grafo-finalizado true
]
[
user-message( "No se puede modificar el grafo una vez finalizado.")
]
]
end

;Método para colorear el interior del grafo.
to colorear-interior
  while [any? patches with [pcolor = white and any? neighbors4 with [pcolor
= black]]]
```

```

[
  ask patches with [pcolor = white and any? neighbors4 with [pcolor =
black]]
  [
    ask neighbors4 with [pcolor = black] [set pcolor white]
  ]
]
end

;Método para conocer los recorredores del patch.
to recorredor-del-patch
  if mouse-inside? and mouse-down? != pulsado
  [
    ask patch mouse-xcor mouse-ycor
    [
      if((is-list? recorredores-vistos) and (length recorredores-vistos >0))
      [
        sprout-observadores 1
        [
          setxy xcor ycor
          set color red
          set size 3
          set shape circle"
        ]
        foreach recorredores-vistos
        [ [?1] ->

```

---

```
ask recorredor ?1
[
set color blue - 1
set shape circle"
set size 2
]
]
set contador-recorredores-vistos (count recorredores with [color = blue -
1])
;output-show(word Coordenadas : "mouse-xcor " , "mouse-ycor)
]
wait .2
]
]
set pulsado mouse-down?
end

;Método para borrar los observadores.
to anula-recorredor-del-patch
ask observadores [die]
ask recorredores [set color orange]
set contador-recorredores-vistos 0
clear-output
end

;Método para dibujar los patches del foco.
to colorea-patches
```

```

let maximo-valor-observado 0
ask max-one-of patches [valor-observado]
[
set maximo-valor-observado valor-observado
]
ask patches with [valor-observado >0]
[
set pcolor (9.9 * (valor-observado / (maximo-valor-observado + 1)))
]
end
;----- Parámetros -----
;Método para generar los observadores en el entorno.
to-report genera-posicion-aleatoria-correcta
let lista-retorno []
ask one-of patches with[(is-list? recorredores-vistos) and (length
recorredores-vistos >0)]
[
set lista-retorno (list pxcor pycor)
]
report lista-retorno
end
to crea-observadores
selecciona-numero-observadores
create-observadores numero-observadores
[

```

---

```

let coordenadas genera-posicion-aleatoria-correcta
let xtemp item 0 coordenadas
let ytemp item 1 coordenadas
setxy xtemp ytemp
set shape "square"
set color red
set label-color yellow
set size 5
]
end

;Método para seleccionar el número de observadores.
to selecciona-numero-observadores
let numero-observadores-aristas 0
let numero-observadores-trig-rmax 0
let numero-observadores-trig-deg 0
set numero-observadores-aristas (ceiling ((etiqueta-global + 1) / 2)) + 1
set numero-observadores-trig-rmax (ceiling ((contador-recorredores * sqrt
(2)) / (2 * (sqrt ((radio-maximo) ^ 2 - (radio-minimo) ^ 2 )))))
set numero-observadores-trig-deg (ceiling ((contador-recorredores * sqrt
(2)) / (2 * ((radio-minimo / tan(20)))))
let numero-observadores-list []
set numero-observadores-list (list numero-observadores-aristas numero-
observadores-trig-rmax)
set numero-observadores max numero-observadores-list
end

; BEGIN MAGIA NEGRA

```

; Procedimiento de tortuga ; Devuelve los patches visibles en el cono de radio y ángulo especificado ; y que no estén ocultos por la lista de Segmentos

to-report patches-visible-in-cone [radius angle Segments]

report patches in-cone radius angle with [pcolor = black and is-visible? Segments]

end

; Funciones de cálculo de intersección de dos segmentos ; Ver: <http://bryceboe.com/2006/10/23/line-segment-intersection-algorithm/> ; A B C D son tortugas, y se mira si el segmento AB corta al CD

to-report ccw [A B C]

let A.x [xcor] of A

let A.y [ycor] of A

let B.x [xcor] of B

let B.y [ycor] of B

let C.x [xcor] of C

let C.y [ycor] of C

report ((C.y - A.y) \* (B.x - A.x)) > ((B.y - A.y) \* (C.x - A.x))

end

to-report intersect [S C D]

let A [end1] of S

let B [end2] of S

report (ccw A C D) != (ccw B C D) and (ccw A B C) != (ccw A B D)

end

to-report is-visible? [Segments]

; ob contendrá una tortuga temporal sobre el patch actual

let ob 0

sprout 1 [



---

```

set ob self
]
; res contendrá la respuesta final
let res false
; Para cada segmento se mira si la recta de visión desde el vigilante
; (myself) hasta el patch actual (ob) no lo intersecta. Finalmente, hacemos
el and de todas esas respuestas parciales
if reduce and map [ [?1] ->not intersect ?1 myself ob ] Segments
[ set res true ]
; Matamos la tortuga temporal
ask ob [die]
; Devolvemos la respuesta
report res
end
; END MAGIA NEGRA
;Método para activar los focos de los observadores.
to enciende-focos
ask patches with [pcolor != orange and pcolor != white]
[ set pcolor black
set recorredores-vistos [] ]
ask recorredores
[ let idrecorredor who
ask recorredores
[
forward 1

```

```

]
ask patches-visible-in-cone distancia-maxima (2 * angulo-max-incidencia)
(sort links)
[
set valor-observado valor-observado + 1
set recorredorVisto idrecorredor
set recorredores-vistos lput recorredorVisto recorredores-vistos
]
ask recorredores
[
forward -1
]
]
colorea-patches
end
;————— Algoritmo genético —————
; La función inicializa-genetico inicializa el algoritmo genético, borrando
todos los individuos y creando una población aleatoria
to inicializar-genetico
reset-ticks
reset-timer
ask individuos
[
die
]
grabar-individuos-titulo

```

---

```
selecciona-numero-observadores
; Genera la población aleatoria
create-individuos numero-individuos-poblacion
[
genera-individuo-aleatorio
calcular-fitness
hide-turtle
]
ask observadores[die]
ask recorredores
[
set color orange
]
end
; Ejecuta un paso del algoritmo genético
to iteracion-genetica
crea-nueva-generacion
mata-individuos
tick
set clock timer
grabar-individuos
ask observadores[die]
ask recorredores
[
set color orange
```

```

]
let mejor max-one-of individuos [fitness]
ask mejor
[
  let observadores-activados map last filter [ [?1] ->first ?1 = 1 ] (map list
lista-activaciones lista-observadores)
  foreach observadores-activados
  [ [?1] ->
    let posicion-x item 0 ?1
    let posicion-y item 1 ?1
    hatch-observadores 1
    [
      setxy posicion-x posicion-y
      set shape "square"
      set color red
      set label-color yellow
      set size 5
      show-turtle
      ; sprout-observadores 1 [setxy xcor ycor set color red set size 3 set shape
circle"]
      foreach recorredores-vistos [ [??1] ->
        ask recorredor ??1 [
          set color blue - 1
          set shape circle"
          set size 2
        ]
      ]
    ]
  ]

```

---

```

]
set contador-recorredores-vistos (count recorredores with [color = blue -
1])
]
]
]
end

; Genera un individuo aleatorio
to genera-individuo-aleatorio
  set lista-observadores n-values numero-observadores [genera-posicion-
aleatoria-correcta]
  set lista-activaciones n-values numero-observadores [random 2]
  end
  ; :D
  to-report list-intersect [a b]
    report (filter [ [?1] ->member? ?1 b ] a)
  end
  ; Calcula el fitness para un individuo
  to calcular-fitness
    let observadores-activados map last filter [ [?1] ->first ?1 = 1 ] (map list
lista-activaciones lista-observadores)
    let todos-los-observadores []
    let min-observadores 99999999
    let max-solapamiento 0
    foreach observadores-activados
      [ [?1] ->

```

```

let posicion-x item 0 ?1
let posicion-y item 1 ?1
ask patch posicion-x posicion-y
[
  set todos-los-observadores (sentence todos-los-observadores recorredores-
vistos)
  set min-observadores (min (list min-observadores (length recorredores-
vistos)))
]
]

let observadores-sin-repeticion (remove-duplicates todos-los-
observadores)

set fitness (length observadores-sin-repeticion) - 10 * (length
observadores-activados) + min-observadores end

; Crea la siguiente generacion
to crea-nueva-generacion
let generacion-anterior individuos with [true]

let numero-cruces max list 2 (numero-individuos-poblacion * porcentaje-
padres * 0.01 / 2)

; Realiza los cruces
repeat numero-cruces
[ let padre-1 max-one-of (n-of 4 generacion-anterior) [fitness]
let padre-2 max-one-of (n-of 4 generacion-anterior) [fitness]
crea-hijos padre-1 padre-2
]
end

to crea-hijos [padre-1 padre-2]

```

---

```

; Selecciona un punto de corte aleatorio

let punto-corte 1 + random (numero-observadores - 1)

; Crea los observadores y las activaciones de los hijos

let hijo-1-observadores (sentence (sublist ([lista-observadores] of padre-1)
0 punto-corte) (sublist ([lista-observadores] of padre-2) punto-corte numero-
observadores))

let hijo-2-observadores (sentence (sublist ([lista-observadores] of padre-2)
0 punto-corte) (sublist ([lista-observadores] of padre-1) punto-corte numero-
observadores))

let hijo-1-activaciones (sentence (sublist ([lista-activaciones] of padre-1)
0 punto-corte) (sublist ([lista-activaciones] of padre-2) punto-corte numero-
observadores))

let hijo-2-activaciones (sentence (sublist ([lista-activaciones] of padre-2)
0 punto-corte) (sublist ([lista-activaciones] of padre-1) punto-corte numero-
observadores))

; Mutacion

set hijo-1-observadores (muta-observadores hijo-1-observadores)

set hijo-2-observadores (muta-observadores hijo-2-observadores)

set hijo-1-activaciones (muta-activaciones hijo-1-activaciones)

set hijo-2-activaciones (muta-activaciones hijo-2-activaciones)

; Crear hijos

create-individuos 1

[
hide-turtle

set lista-observadores hijo-1-observadores

set lista-activaciones hijo-1-activaciones

calcular-fitness

]

```

```

create-individuos 1
[
hide-turtle

set lista-observadores hijo-2-observadores
set lista-activaciones hijo-2-activaciones
calcular-fitness
]
end

to-report muta-observadores [obs]
  let nuevas-observadores (map [ [?1] ->ifelse-value (random-float 1
<probabilidad-mutacion-gen) [genera-posicion-aleatoria-correcta] [?1] ] obs)
  report nuevas-observadores
end

to-report muta-activaciones [act]
  let nuevas-activaciones (map [ [?1] ->ifelse-value (random-float 1
<probabilidad-mutacion-gen) [1 - ?1] [?1] ] act)
  report nuevas-activaciones
end

; Mata a los individuos con un fitness peor
to mata-individuos
  let numero-individuos (count individuos)
  let numero-muertes (numero-individuos - numero-individuos-poblacion)
  repeat numero-muertes
  [
ask min-one-of (n-of 4 individuos) [fitness]
[

```



---

```

die
]
]
end
to muestra-mejor-individuo
let mejor 0
end
;----- Guarda logs -----
;Método para crear el archivo de logs y ponerle título.
to grabar-individuos-titulo
let nombre-archivo "solucion.csv"
if file-exists? nombre-archivo
[
file-delete nombre-archivo
]
file-open nombre-archivo
let titulo "FITNESS,ITERACIÓN,TIEMPO"
foreach n-values numero-observadores [ [?1] ->?1 ]
[ [?1] ->
set titulo (word titulo ",X"(?1 + 1) ",Y"(?1 + 1))
]
file-print titulo
file-close
end
;Método para rellenar el archivo de logs con los datos del algoritmo gené-
```

```

tico

  to grabar-individuos

    let nombre-archivo "solucion.csv"

    file-open nombre-archivo

    let mejor max-one-of individuos [fitness]

    ask mejor

    [

      let observadores-activados map last filter [ [?1] ->first ?1 = 1 ] (map list
lista-activaciones lista-observadores)

      let texto-observadores

      foreach observadores-activados

        [ [?1] ->

          set texto-observadores (word texto-observadores ","(item 0 ?1) ","(item 1
?1))

        ]

      file-print (word fitness ","ticks ",clock texto-observadores)

    ]

    file-close

  end

  to cargar-csv

    set point-list but-first but-last (csv:from-file nombre-csv ",")

    ; Creando las 3 sublistas a partir de la lista del CSV.

    let point-list-x []

    let point-list-y []

    let point-list-id []

    foreach point-list

```

---

```
[ [?1] ->
let temp0 (item 0 ?1)
let temp1 (item 1 ?1)
let temp2 (item 2 ?1)
set point-list-x lput temp0 point-list-x
set point-list-y lput temp1 point-list-y
set point-list-id lput temp2 point-list-id
]
; Calcular los máximos y mínimos de la nube de puntos
let min-x 9999999999999999
let min-y 9999999999999999
let max-x -9999999999999999
let max-y -9999999999999999
foreach point-list
[ [?1] ->
let temp0 (item 0 ?1)
let temp1 (item 1 ?1)
set max-x max (list max-x temp0)
set max-y max (list max-y temp1)
set min-x min (list min-x temp0)
set min-y min (list min-y temp1)
]
; Centrar el mundo
let centro-x ((min-x + max-x) / 2)
let centro-y ((min-y + max-y) / 2)
```

```

let lista-modificada-puntos-centro-x []

let lista-modificada-puntos-centro-y []

foreach point-list-x
[ [?1] ->
  set lista-modificada-puntos-centro-x lput ((?1) + (centro-x * (-1))) lista-
modificada-puntos-centro-x
]

foreach point-list-y
[ [?1] ->
  set lista-modificada-puntos-centro-y lput ((?1) + (centro-y * (-1))) lista-
modificada-puntos-centro-y
]

; Escalando el mundo

let dist-mund-rel-x (max-x - min-x)

let dist-mund-rel-y (max-y - min-y)

let max-mund-anchura (world-width - 1) * 0.8

let max-mund-altura (world-height - 1) * 0.8

let factor-y (max-mund-altura / dist-mund-rel-y)

let factor-x (max-mund-anchura / dist-mund-rel-x)

let min-condicionante min (list (factor-y) (factor-x))

let lista-escalada-puntos-x []

let lista-escalada-puntos-y []

foreach lista-modificada-puntos-centro-x
[ [?1] ->
  set lista-escalada-puntos-x lput ((?1) * min-condicionante) lista-escalada-
puntos-x

```

---

```
]
foreach lista-modificada-puntos-centro-y
[ [?1] ->
  set lista-escalada-puntos-y lput ((?1) * min-condicionante) lista-escalada-
puntos-y
]
let cont 0
let len (length lista-escalada-puntos-x)
while [cont <len]
[
  let x-temp item cont lista-escalada-puntos-x
  let y-temp item cont lista-escalada-puntos-y
  aerodinamico x-temp y-temp
  set cont (cont + 1)
]
finalizar-grafo
end
```



# Índice de figuras

1.1. Fuente Pabellón de Marruecos de 1929, Sevilla. Modelo obtenido a través de técnicas de Fotogrametría (Ornia Núñez et al., 2017). . . . .	3
1.2. Modelo tridimensional de la fortaleza de Al Zubarah, Qatar, generada con el <i>software Agisoft PhotoScan</i> (Kersten et al., 2015). .	6
1.3. Capitel impreso en 3D. Elaboración Propia. . . . .	6
1.4. Optimización por enjambre de partículas (Sancho Caparrini, 2015). . . . .	8
1.5. Complejidad computacional de los problemas (Sancho Caparrini, 2009). . . . .	11
1.6. Modelo tridimensional de fragmento de escultura de mármol perteneciente al Museo Arqueológico de Paestum (Italia). Elaboración Propia. . . . .	14
1.7. Maqueta de Miguel Ángel para la cúpula de San Pedro del Vaticano (Almagro Gorbea, 2004, p.35). . . . .	15
1.8. Nube de puntos del Segundo Templo de Hera, Paestum (Italia). Elaboración Propia. . . . .	15
1.9. Instrumentos simples de medición (Almagro Gorbea, 2004, p.42). .	18
1.10. Estación Topográfica de última generación. Fuente: Leica Geosystem. . . . .	18
1.11. Trabajo de Fotogrametría realizado con el <i>software Agisoft Photoscan</i> (Ramírez Pérez and Cabrera Revuelta, 2016). . . . .	19

1.12. Escáner Láser de última generación. Fuente: Leica Geosystem. .	20
1.13. Reconstrucción de rayos de visión para obtención del punto en el espacio (Tang et al., 2012). . . . .	22
1.14. Interpretación del cerebro debido a la superposición de imágenes tomadas desde posiciones levemente desviadas (Tucci et al., 2013). . . . .	23
1.15. Esquema de Proyección Perspectiva (Marshall, 1989). . . . .	25
1.16. Principio de la Condición de Colinealidad (Russo et al., 2011). .	26
1.17. Principio de la triangulación fotogramétrica o <i>Bundle Adjustment</i> (Russo et al., 2011). . . . .	27
1.18. Esquema de funcionamiento de un escáner láser de triangulación (Russo et al., 2011). . . . .	30
1.19. Resolución de un escaneo (Nobile, 2012). . . . .	31
1.20. A igualdad de resolución angular, la densidad de puntos varía en función a la inclinación de la superficie con respecto al instrumento (Nobile, 2012). . . . .	32
1.21. ZEB1 utilizado en el yacimiento arqueológico de Baia (Italia). Elaboración Propia. . . . .	32
1.22. ZEB-REVO con accesorios. Fuente: GeoSLAM. . . . .	33
1.23. Galería enterrada, Trento (Italia) (Rodríguez-Gonzálvez et al., 2015). . . . .	34
1.24. Nube de puntos de los 19 escaneos (Rodríguez-Gonzálvez et al., 2015). . . . .	35
1.25. Reconstrucción de la fortaleza (Yabe et al., 2015). . . . .	35
1.26. Hipótesis de reconstrucción del castillo mostrando las ruinas existentes integradas en él (Camillo Saviello et al., 2017). . . .	36
1.27. Vista del Castillo y su entorno simulando una reconstrucción de 1629 (Camillo Saviello et al., 2017). . . . .	37
1.28. Reconstrucción interactiva del <i>agora</i> (Olivito et al., 2015). . . .	37
1.29. Fotocubrición Estereoscópica (CIPA, 2013). . . . .	38



1.30.Par Estereoscópico (CIPA, 2013). . . . .	39
1.31.El comportamiento de la colonia termina por obtener el camino más corto entre dos puntos (Sancho Caparrini, 2015). . . . .	50
1.32.Distribución de publicaciones científicas usando las distintas metaheurísticas (Fawzy, 2015, p.148). . . . .	50
1.33.La medida del eje menor y del eje mayor del elipsoide que se forma, es una medida de calidad de la geometría de la escena (Beder and Steffen, 2006). . . . .	53
1.34.Representación de Error Elipsoidal (Olague, 2001). . . . .	53
1.35.Eschema de Algoritmo Genético (Olague, 2001, p.428). . . . .	55
1.36.Evaluación de una red de fotogrametría (Olague and Dunn, 2007). . . . .	56
1.37.Objeto tridimensional y su correspondiente grafo (Roberts and Marshall, 1998). . . . .	57
1.38.Vistas reales y esquema de las caras vigiladas para cada grupo. Sólo las caras que faltan por ser vistas son representadas en el esquema (Roberts and Marshall, 1998). . . . .	58
1.39.(a) Un mal punto de vista y (b) un buen punto de vista de una escena que contiene una esfera y un cubo (Vázquez et al., 2001). . . . .	59
1.40.Los ocho puntos de vista obtenidos para visualizar la escena (Vázquez et al., 2001). . . . .	59
1.41.Micro dron md4-200 con cámara anexa (Irschara et al., 2010, p.65). . . . .	60
1.42.Clasificación de las cámaras según el ángulo (Hoppe et al., 2012, p.3). . . . .	62
1.43.Ejemplo de un Histograma de un triángulo T con cuatro grupos. En a) la caja sombreada marca la que ve al mayor número de triángulos. En b) se han eliminado las cámaras del primer grupo tras haber seleccionado la $C_{max}$ (Hoppe et al., 2012, p.3). . . . .	63
1.44.Ejemplo de cámaras utilizadas tras aplicar el Algoritmo (Hoppe et al., 2012, p.6). . . . .	64

1.45. Cámaras utilizadas en color gris, cámaras no usadas en color rojo, y camino en color amarillo (Hoppe et al., 2012, p.1).	65
1.46. a) Red densa, b) Red mínima de cubrición, c) Red mejorada (Alsadik et al., 2012, p.11).	67
1.47. Vista esquemática de un AOI, las cadenas de vuelo, y las vueltas (Dashora et al., 2013).	68
1.48. Región de búsqueda de puntos de vista (Yang et al., 2014).	70
1.49. Generación de puntos de vista candidatos. La región local del punto de vista $k$ es una parte de una superficie cilíndrica (Yang et al., 2014).	70
1.50. Representación del vector normal de la superficie, $n_k$ y el vector la cámara, $v_i$ (Yang et al., 2014).	71
1.51. Imágenes capturadas por puntos de vista candidatos (Yang et al., 2014).	72
1.52. Robot 3DIrma (Borrmann et al., 2015).	73
1.53. Planta escaneada del Salón Imperial del Palacio Würzburg y la ruta realizada (Borrmann et al., 2015).	74
1.54. Principio de funcionamiento del complemento <i>Viewshed Analysis</i> de QGIS (Alonso, 2016).	75
1.55. Aplicación de la herramienta <i>Viewshed Analysis</i> de QGIS (Rosser et al., 2017).	75
4.1. El peine de Chvátal (O'Rourke, 1998, p.6).	88
4.2. Polígono triangulado, 3-coloreado y su gráfica dual (Morales, 2006).	90
4.3. Algoritmo de Avis y Toussaint (Morales, 2006).	91
4.4. Construcción de los canales (Bjorling-Sachs and Souvaine, 1995).	93
4.5. Triangulación del nuevo polígono con $n + h$ vértices (Bjorling-Sachs and Souvaine, 1995).	93
4.6. Polígono ortogonal cuadrilaterizado, su gráfica $H$ y 4 coloreado. (Morales, 2006).	94

4.7. a) Triangulación de polígono ortogonal; b) Unión de pares de triángulos no dan como resultados cuadriláteros convexos; c) Correcta cuadrilaterización convexa (O'Rourke, 1987). . . . .	95
4.8. Polígono con 2 obstáculos, triangulado y 3-coloreado (Morales, 2006). . . . .	96
4.9. El Grafo $G$ en c) es producido desde la triangulación de las bolsas en a) y conectando todos los vértices del cierre convexo a $v_\infty$ , en b), se separa algún vértice $x$ en dos: $x'$ y $x''$ (O'Rourke, 1987). . . . .	97
4.10. Pasos del Algoritmo del Problema de la Fortaleza (O'Rourke, 1987). . . . .	98
4.11. El exterior de un polígono ortogonal debe ser convertido en el interior de otro polígono eliminando la arista más alta $e$ y encerrándolo en un rectángulo (O'Rourke, 1987). . . . .	100
4.12. Prueba de que $\lceil n/3 \rceil$ guardias son necesarios para cubrir el exterior de esta figura (O'Rourke, 1987, p.151). . . . .	100
4.13. Guardias en el exterior del polígono (O'Rourke, 1987, p.151). . . . .	101
4.14. Visibilidad de alcance $d$ (Canales Cano, 2004). . . . .	102
4.15. Cada arista es dividida en celdas, y a cada punto de vista se le asignan las celdas que vigila correctamente (González-Baños, 2001). . . . .	105
4.16. a) Tres guardias son suficientes para vigilar el contorno interior del polígono sin limitación del ángulo de incidencia. b) Seis guardias son necesarios cuando el ángulo de incidencia está limitado (González-Baños, 2001). . . . .	106
4.17. Gráfico de Flujo de un Algoritmo Genético. Elaboración Propia, Fuente (Majeed and Kumar, 2014). . . . .	110
4.18. Esquema de los elementos que componen un Algoritmo Genético. Elaboración Propia. . . . .	112
4.19. Selección por Ruleta (Ramírez, 2008). . . . .	115
4.20. Cruce en un punto (Gestal et al., 2010, p.21). . . . .	117
4.21. Cruce en dos puntos (Gestal et al., 2010, p.22). . . . .	117

4.22. Cruce Uniforme (Gestal et al., 2010, p.23). . . . .	118
4.23. Adaptación media y mejor adaptación en un Algoritmo Genético (Moujahid et al., 2008). . . . .	120
4.24. Un individuo $I$ de un polígono arbitrario de $n = 20$ vértices (Bajuelos Domínguez et al., 2008). . . . .	121
5.1. Aplicación del algoritmo del Problema de la Fortaleza sobre un polígono. Elaboración Propia. . . . .	125
5.2. Polígono $P$ con vértices y aristas nombradas. Elaboración Propia.	126
5.3. Cierre convexo. Elaboración Propia. . . . .	127
5.4. Triangulación de las bolsas. Elaboración Propia. . . . .	128
5.5. Nuevo vértice $v_{\infty}$ y triangulación del exterior de $P$ mediante la unión de cada vértice del cierre convexo con $v_{\infty}$ . Elaboración Propia. . . . .	129
5.6. Apertura de $P$ en un vértice $v_i$ perteneciente al cierre convexo, creándose $v_{i'}$ , y unión de este nuevo vértice con $v_{\infty}$ . El inte- rior del polígono $P$ es ahora el exterior de <i>polígono</i> $P'$ , el cual aparece sombreado. Elaboración Propia. . . . .	130
5.7. Bicoloración de los vértices del cierre convexo y el vértice $v_{i'}$ , comenzando por éste vértice y continuando en sentido horario. Elaboración Propia. . . . .	131
5.8. 3-coloración de los vértices y triangulación del espacio. Elabo- ración Propia. . . . .	132
5.9. Polígono $P$ con cierre convexo, y triangulación realizada, apre- ciándose los vértices de color rojo y las aristas que cada uno tiene por misión vigilar. Elaboración Propia. . . . .	133
5.10. Área de visibilidad de la arista $a_1$ , a través del trazado de la recta $R_1$ y obtención del semiplano $\alpha_{r_1}$ . Elaboración Propia. . . . .	134
5.11. Área de visibilidad de la arista $a_2$ . Elaboración Propia. . . . .	135
5.12. Intersección de áreas de visibilidad de las aristas $a_1, a_2, a_3, a_4$ y $a_5$ para la obtención del área de visibilidad de $v_2$ , la cual se refleja más sombreada. Elaboración Propia. . . . .	136

5.13. Intersección de áreas de visibilidad de las aristas $a_6$ y $a_7$ para la obtención del área de visibilidad de $v_7$ , la cual se refleja más sombreada. Elaboración Propia. . . . .	138
5.14. Intersección de áreas de visibilidad de las aristas $a_8, a_9$ y $a_{10}$ para la obtención del área de visibilidad de $v_{10}$ , la cual se refleja más sombreada. Elaboración Propia. . . . .	139
5.15. Área de visibilidad de $v_{11}$ . Elaboración Propia. . . . .	140
5.16. Área de visibilidad de $v_{13}$ . Elaboración Propia. . . . .	142
5.17. Área de visibilidad de $v_{16}$ . Elaboración Propia. . . . .	143
5.18. Representación de todas las áreas de visibilidad del subconjunto de vértices de color rojo. Obsérvese que la zona más sombreada es debido al solape de dos áreas de visibilidad, la de $v_{13}$ y la de $v_{16}$ . Elaboración Propia. . . . .	144
5.19. Polígono ortogonal, P Elaboración Propia. . . . .	145
5.20. Polígono ortogonal, P, encerrado en recuadro. Elaboración Propia. . . . .	146
5.21. Eliminación de la arista superior $e$ . Elaboración Propia. . . . .	147
5.22. Extensión de aristas adyacentes a $e$ , creación de polígono $P'$ de $n + 4$ vértices, y rectángulo Q. Elaboración Propia. . . . .	148
5.23. Cuadrilaterización de $P'$ . Elaboración Propia. . . . .	149
5.24. 4-Coloración de los vértices de $P'$ . Elaboración Propia. . . . .	150
5.25. Áreas de visibilidad obtenidas para garantizar la vigilancia de P. Elaboración Propia. . . . .	151
5.26. Pantalla de representación de datos (Martín López, 2015b). . . . .	152
5.27. Prueba sobre polígono de forma simple (Martín López, 2015b). . . . .	153
5.28. Prueba sobre polígono de forma compleja (Martín López, 2015b). . . . .	154
5.29. Prueba sobre polígono de forma compleja 2 (Martín López, 2015b). . . . .	154
5.30. Prueba sobre polígono de forma compleja 3 (Martín López, 2015b). . . . .	155

5.31.Prueba sobre polígono de forma compleja 4 (Martín López, 2015b). . . . .	155
5.32.Prueba sobre polígono de forma compleja 5 (Martín López, 2015b). . . . .	156
5.33.Posible solución, o individuo. Elaboración Propia. . . . .	158
5.34.En el ejemplo expuesto, existen cuatro componentes. Se observa que de las rectas trazadas, sólo una no atraviesa el edificio. El individuo constituido por cuatro componentes tiene por valor $k_c = 1$ . Elaboración Propia. . . . .	164
5.35.Edificio siendo vigilado por el individuo $I_1$ . Elaboración Propia. . . . .	167
5.36.Edificio siendo vigilado por el individuo $I_2$ . Elaboración Propia. . . . .	168
5.37.Edificio siendo vigilado por el individuo $I_3$ . Elaboración Propia. . . . .	170
5.38.Subconjuntos de individuos de la población inicial tras el proceso de Selección. Elaboración Propia. . . . .	171
5.39.Dos individuos $I_1$ e $I_2$ compuestos por sus respectivos componentes, en azul y en negro. Elaboración Propia. . . . .	172
5.40.División del espacio en $k = 6$ sectores. El nuevo individuo está compuesto por seis posicionamientos. Elaboración Propia. . . . .	173
5.41.Los componentes del nuevo individuo se crean de distintas formas: $C_1$ , $C_2$ , $C_5$ y $C_6$ se ubican en la misma posición donde estuvo un componente de alguno de sus antecesores. $C_3$ se crea de forma aleatoria. $C_4$ es combinación lineal de sus antecesores. Elaboración Propia. . . . .	174
5.42.Área de visión emitida por cada uno de los recorredores en un <i>patch</i> determinado P. Elaboración Propia. . . . .	182
5.43.Área de visión de un punto P limitada por obstáculos que limitan la visibilidad completa. Elaboración Propia. . . . .	183
5.44.Interfaz del modelo diseñado en <i>Net-Logo</i> . Elaboración Propia. . . . .	188
5.45.Dibujar los vértices en el sentido deseado. Elaboración Propia. . . . .	189
5.46.Seperación del espacio en tres regiones: interior del polígono, exterior del polígono y borde del polígono. Elaboración Propia. . . . .	190

5.47. Terreno inspeccionado. Elaboración Propia. . . . .	191
5.48. Opcional: Pinchar sobre posibles puntos de vista para analizar la visibilidad de los mismos. Elaboración Propia. . . . .	192
5.49. Solución obtenida a través del AG. Con seis posicionamientos es posible cubrir la vigilancia completa del polígono. Elaboración Propia. . . . .	193
5.50. Terreno inspeccionado en Caso I. Elaboración Propia. . . . .	194
5.51. Solución obtenida tras aplicar el AG, Caso I. Elaboración Propia. . . . .	195
5.52. Evolución del Valor de Ajuste y del número de componentes, Caso I. Elaboración Propia. . . . .	196
5.53. Solución obtenida tras aplicar el AG, Caso II. Elaboración Propia. . . . .	197
5.54. Evolución del Valor de Ajuste y del número de componentes, Caso II. Elaboración Propia. . . . .	198
5.55. Solución obtenida tras aplicar el AG, Caso III. Elaboración Propia. . . . .	199
5.56. Evolución del Valor de Ajuste y del número de componentes, Caso III. Elaboración Propia. . . . .	200
5.57. Solución obtenida tras aplicar el AG, Caso IV. Elaboración Propia. . . . .	201
5.58. Evolución del Valor de Ajuste y del número de componentes, Caso IV. Elaboración Propia. . . . .	202
A.1. Distribución geográfica de los molinos de marea (Ménanteau and Pourinet, 2005). . . . .	213
A.2. Esquema de funcionamiento de un molino de marea (Azurmen- di Pérez, 2011). . . . .	214
A.3. Parque Natural de la Bahía de Cádiz (Machuca Macías, 2013). . . . .	217
A.4. Vista aérea del Parque Natural de la Bahía de Cádiz. Fuente (Ins- tituto Geográfico Nacional, 2017). . . . .	218
A.5. Vista aérea del Parque Natural de la Bahía de Cádiz. Fuente (Ins- tituto Geográfico Nacional, 2017). . . . .	219

A.6. Bajada del nivel del mar en la Bahía de Cádiz (Machuca Macías, 2013). . . . .	219
A.7. Resumen de características de los molinos de marea del Parque Natural de la Bahía de Cádiz. Elaboración propia. Fuente (Molina Font, 2001). . . . .	220
A.8. Ubicación de los molinos de marea de la Bahía de Cádiz (Mé-nanteau and Pourinet, 2002). . . . .	221
A.9. Cádiz y el Parque Natural (Machuca Macías, 2013). . . . .	222
A.10. Las Salinas de San Félix (Machuca Macías, 2013). . . . .	223
A.11. Imagen de hipótesis de la forma en L primitiva del molino (Tro-cadero, 2011). . . . .	224
A.12. Distribución interna del molino (Machuca Macías, 2013). . . . .	224
A.13. Estado actual. Planta baja (Machuca Macías, 2013). . . . .	225
A.14. Estado actual. Planta cubierta (Machuca Macías, 2013). . . . .	226
A.15. Estado actual. Interior. Elaboración Propia. . . . .	227
A.16. Estado actual. Interior. Elaboración Propia. . . . .	227
A.17. Estado actual. Exterior. Elaboración Propia. . . . .	228
A.18. Dibujo del molino hecho <i>in situ</i> con plumilla rellena de tinta china negra y con acuarela <i>Windsor and Newton</i> (Alcina Segura, 2016, p.199). . . . .	228
A.19. Creación de capa vectorial en QGIS para la digitalización de la planta del edificio sobre ortofoto de la zona. Elaboración Propia. . . . .	230
A.20. Puntos de vista obtenidos tras la aplicación del AG. Elaboración Propia. . . . .	231
A.21. Cámara panorámica iSTAR 360° de Nctech. Elaboración Propia. . . . .	231
A.22. Nubes de puntos registradas y posición de cada uno de los esca-neos. Elaboración Propia. . . . .	232
A.23. Alzado norte visualizado desde <i>Cyclone</i> . Elaboración Propia. . . . .	233
A.24. Alzado sur visualizado desde <i>Cyclone</i> . Elaboración Propia. . . . .	233



A.25. Vista en perspectiva. Elaboración Propia. . . . .	234
A.26. Georreferenciación de la nube de puntos a través del método Helmert. Elaboración Propia. . . . .	234
B.1. Variación del ángulo $\gamma$ . Elaboración Propia. . . . .	235
B.2. Resultados obtenidos en el análisis de precisiones del distanció- metro láser sobre diferentes materiales. Gráficos de respuesta del azulejo y del ladrillo con estación total computerizada Leica TPS 700 (García et al., 2013). . . . .	236
B.3. Lectura de paramento de mármol y dianas colocadas. Elabora- ción Propia. . . . .	238
B.4. Lectura de paramento de fábrica de ladrillo visto con escáner láser. Elaboración Propia. . . . .	239
B.5. Nube de puntos dispersa del paramento FLV. Rayos lanzados des- de el escáner a cada una de las dianas, así como superficies pla- nas creadas alrededor de cada una de las dianas. Elaboración Propia. . . . .	241
B.6. Análisis de la variación de la densidad de puntos alrededor de cuatro puntos ubicados en ángulos determinados ( $0^\circ$ , $45^\circ$ , $60^\circ$ , $70^\circ$ ), y en tres tipos de paramentos distintos (M, FLV y HV). Elaboración Propia. . . . .	242
B.7. Variación del error en tres tipos de paramentos distntos (M, FLV y HV), en función al valor del $\gamma$ . Elaboración Propia. . . . .	242



# Referencias

- Abellanas, M. and Hernández, G. (2007). Optimización de rutas de evacuación. *Proceedings of the XII Encuentros de Geometría Computacional, Valladolid, Spain*, pages 273–280.
- Ackley, D. (2012). *A Connectionist Machine for Genetic Hillclimbing*. The Springer International Series in Engineering and Computer Science. Springer US.
- Aggarwal, A. (1984). *The Art Gallery Theorem: Its Variations, Applications and Algorithmic Aspects*. PhD thesis.
- Alander, J. T. (1992). On Optimal Population Size of Genetic Algorithms. In *CompEuro'92. 'Computer Systems and Software Engineering', Proceedings*, pages 65–70. IEEE.
- Alcina Segura, J. (2016). *La provincia de Cádiz en un cuaderno: Arquitectura - Urbanismo - Paisaje - Historia*. Q-book Cultura Integral, S.L.
- Almagro Gorbea, A. (2004). *Levantamiento arquitectónico*. Universidad de Granada.
- Alonso, D. (2016). Cómo realizar un análisis de visibilidad con QGIS. <https://mappinggis.com/2016/02/como-realizar-un-analisis-de-visibilidad-con-qgis/>.
- Alsadik, B., Gerke, M., and Vosselman, G. (2012). Optimal Camera Network Design for 3D Modeling of Cultural Heritage. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3:7–12.
- Appel, K., Haken, W., and Koch, J. (1977). Every planar map is four colorable. Part II: Reducibility. *Illinois J. Math.*, 21(3):491–567.
- Avis, D. and Toussaint, G. T. (1981). An optimal algorithm for determining the visibility of a polygon from an edge. *IEEE Trans. Comput.*, 30(12):910–914.

- Azurmendi Pérez, L. (2011). Molinos de marea. *Fabrikart*, (5).
- Bajuelos Domínguez, A. L., Canales Cano, S., Hernández Peñalver, G., and Martins, A. M. (2008). Optimizing the minimum vertex guard set on simple polygons via a genetic algorithm. *Wseas Transactions on Information Science and Applications*, 5(11):1584–1596.
- Barrera Vera, J. A. (2006). *Aplicación de Tecnologías Innovadoras en la Documentación Geométrica del Patrimonio Arquitectónico y Arqueológico*. PhD thesis, Escuela Universitaria de Arquitectura Técnica. Universidad de Sevilla, Sevilla.
- Beder, C. and Steffen, R. (2006). *Determining an Initial Image Pair for Fixing the Scale of a 3D Reconstruction from an Image Sequence*, pages 657–666. Springer Berlin Heidelberg.
- Biedl, T., Chan, T. M., Lee, S., Mehrabi, S., Montecchiani, F., and Vosoughpour, H. (2017). *On Guarding Orthogonal Polygons with Sliding Cameras*, pages 54–65. Springer International Publishing, Cham.
- Bjorling-Sachs, I. and Souvaine, D. L. (1995). An efficient algorithm for guard placement in polygons with holes. *Discrete & Computational Geometry*, 13(1):77–109.
- Blickle, T. and Thiele, L. (1996). A comparison of selection schemes used in evolutionary algorithms. *Evol. Comput.*, 4(4):361–394.
- Borrmann, D., Heß, R., Houshiar, H., Eck, D., Schilling, K., and Nüchter, A. (2015). Robotic mapping of cultural heritage sites. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:9–16.
- Bramlette, M. F. (1991). Initialization, mutation and selection methods in genetic algorithms for function optimization. In *Fourth International Conference on Genetic Algorithms*, pages 100–107.
- Brenner, W., Wittig, H., and Zarnekow, R. (1998). *Intelligent Software Agents: Foundations and Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1st edition.
- Brown, D. C. (1958). *A Solution to the General Problem of Multiple Station Analytical Stereotriangulation*. RCA Data reduction technical report. D. Brown Associates, Incorporated.

- Bryan, P, Blake, B., Bedford, J., Barber, D., and Mills, J. (2013). *Metric Survey Specifications for Cultural Heritage*. English Heritage Publishing.
- Cabrera Revuelta, E., Chávez de Diego, M., and Barrera Vera, J. A. (2010). Optimización en el posicionamiento de estaciones topográficas en los levantamientos de edificios. Master's thesis.
- Cabrera Revuelta, E., Chávez de Diego, M., and Barrera Vera, J. A. (2011). Aplicación del problema de la galería de arte a levantamientos arquitectónicos. In *VII Encuentro Andaluz de Matemática Discreta*, pages 133–135. Universidad de Valencia.
- Cabrera Revuelta, E., Chávez de Diego, M., Barrera Vera, J. A., Caballero Sánchez, M., and Fernández Rodríguez, Y. (2017). Planificación de un levantamiento arquitectónico a través de un algoritmo genético. In *Dibujar, Construir, Soñar*, chapter 3. Tirant lo Blanch.
- Cabrera Revuelta, E., Chávez de Diego, M., Barrera Vera, J. A., and Mole-ro Alonso, B. (2012). Levantamiento de un edificio mediante la aplicación del problema de la galería de arte. In *XI Congreso Internacional de Expresión Gráfica Aplicada a la Edificación*, pages 598–613. Universidad de Valencia.
- Cabrera Revuelta, E., Martín López, F., Chávez de Diego, M. J., Barrera Vera, J. A., Caballero Real, L., Caballero Sánchez, M., Calzada Infante, L., Fernández Rodríguez, Y., and Manjón Velázquez, J. C. (2015a). Genetic algorithm for close range photogrammetric network design. In *Forma 15. Agent Based Modeling*, pages 14–22.
- Cabrera Revuelta, E., Martín López, F., Chávez de Diego, M. J., Barrera Vera, J. A., and Márquez Pérez, A. (2015b). Localización de estacionamientos para la realización de un levantamiento arquitectónico mediante la aplicación de algoritmos genéticos. In *Avances en Matemática Discreta en Andalucía*, pages 59–67.
- Camillo Saviello, S., Barrera Vera, J. A., Barba, S., and Cabrera Revuelta, E. (2017). Levantamiento y Reconstrucción Virtual del Castillo de San Jorge. In *Dibujar, Construir, Soñar*, pages 73–84. Tirant lo Blanch.
- Canales Cano, S. (2004). *Métodos Heurísticos en Problemas Geométricos Visibilidad, Iluminación y Vigilancia*. PhD thesis, Universidad Politécnica de Madrid.
- Chvatal, V. (1975). A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory, Series B*, 18(1):39–41.

- CIPA (2013). The CIPA 3x3 Rules. <http://cipa.icomos.org>.
- Crawford, W. G. (1995). Construction surveying and layout. *Safety*, 3:2.
- CulturaDigital (2015). Jornadas Forma. <http://forma.culturadigital.cc/es/portada>.
- Danaraj, G. and Klee, V. (1974). Shellings of spheres and polytopes. *Duke Math. J.*, 41(2):443–451.
- Dashora, A., Lohani, B., and Deb, K. (2013). Solving flight planning problem for airborne lidar data acquisition using single and multi-objective genetic algorithms. *KanGAL Report No*, 2013012.
- Dashora, A., Lohani, B., and Deb, K. (2014). Method of flight planning for airborne lidar using genetic algorithms. *Journal of Applied Remote Sensing*, 8(1):083576–083576.
- de Frutos, A. M., Díaz, S. J. Y., and Aguilera, D. G. (2011). Aplicaciones de geotecnologías láser y fotogramétricas a la digitalización de escenarios complejos: Castillo de Turégano (Segovia). *Topografía y cartografía: Revista del Ilustre Colegio Oficial de Ingenieros Técnicos en Topografía*, 28(162):24–31.
- de Rezende, P. J., de Souza, C. C., Friedrichs, S., Hemmer, M., Kröller, A., and Tozoni, D. C. (2016). Engineering art galleries. In *Algorithm Engineering*, pages 379–417. Springer.
- Fawzy, H. E.-D. (2015). Comparison between the genetic algorithms optimization and particle swarm optimization for design the close range photogrammetry network. *International Journal of Civil Engineering*, 6(6).
- Fisk, S. (1978). A short proof of Chvátal’s Watchman Theorem. *Journal of Combinatorial Theory, Series B*, 24(3):374.
- Fogarty, T. C. (1989). Varying the probability of mutation in the genetic algorithm. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 104–109, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Fraser, C. (1996). Network design. *Close range photogrammetry and machine vision*, pages 256–281.
- Fraser, C. S. (1984). Network design considerations for non-topographic photogrammetry. *Photogrammetric Engineering and Remote Sensing*, 50(8):1115–1126.

- García, J. A. S., Solís, P. G.-P., Fraga, R. A., and Fernández, B. R. (2013). Topometría láser 3d y cartografía inversa en el tómbolo de Gijón. *Mapping*, (157):20–32.
- Gestal, M., Rivero, D., Rabuñal, J. R., Dorado, J., and Pazos, A. (2010). *Introducción a los algoritmos genéticos y la programación genética*. Universidad da Coruña, Servicio de Publicacións.
- Gigerenzer, G. and Gaissmaier, W. (2011). Heuristic decision making. *Annual review of psychology*, 62:451–482.
- Golberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Addion wesley*, 1989.
- González-Baños, H. (2001). A randomized art-gallery algorithm for sensor placement. In *Proceedings of the seventeenth annual symposium on Computational geometry*, pages 232–240. ACM.
- Grafarend, E. (1974). Optimization of geodetic networks. *Bollentino di Geodeia a Aciencia Affini*. Vol 33, pages 351–406.
- Grimm, V., Berger, U., DeAngelis, D. L., Polhill, J. G., Giske, J., and Railsback, S. F. (2010). The odd protocol: a review and first update. *Ecological modelling*, 221(23):2760–2768.
- Hoffmann, F., Kaufmann, M., and Kriegel, K. (1991). The art gallery theorem for polygons with holes. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*, pages 39–48. IEEE.
- Hoffmann, F. and Kriegel, K. (1996). A graph-coloring result and its consequences for polygon-guarding problems. *SIAM Journal on Discrete Mathematics*, 9(2):210–224.
- Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA.
- Hoppe, C., Wendel, A., Zollmann, S., Pirker, K., Irschara, A., Bischof, H., and Kluckner, S. (2012). Photogrammetric camera network design for micro aerial vehicles. In *Computer vision winter workshop (CVWW)*, volume 8, pages 1–3.
- Instituto Geográfico Nacional, I. (2017). Instituto Geográfico Nacional, Ministerio de Fomento. Gobierno de España. <http://www.ign.es/web/ign/portal>.

- Irschara, A., Kaufmann, V., Klopschitz, M., Bischof, H., and Leberl, F. (2010). Towards fully automatic photogrammetric reconstruction using digital images taken from uavs. In *In Proc. International Society for Photogrammetry and Remote Sensing Symposium*.
- Kahn, J., Klawe, M., and Kleitman, D. (1983). Traditional galleries require fewer watchmen. *SIAM Journal on Algebraic Discrete Methods*, 4(2):194–206.
- Kersten, T., Mechelke, K., and Maziull, L. (2015). 3d model of al Zubarah fortress in Qatar—terrestrial laser scanning vs. dense image matching. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:1–8.
- Kiamehr, R. (2003). Multi object optimization of geodetic network. *NCC Geomatics*, 82.
- Laurentini, A. (1999). Guarding the walls of an art gallery. *The Visual Computer*, 15(6):265–278.
- Lee, D.-T. and Lin, A. K. (1986). Computational complexity of art gallery problems. *Information Theory, IEEE Transactions on*, 32(2):276–282.
- ltdi, F. D. (2017). Geoslam. <http://geoslam.com/>.
- Machuca Macías, C. (2013). Rehabilitación del molino del rio arillo. pfc arquitectura.
- Majeed, P. G. and Kumar, S. (2014). Genetic algorithms in intrusion detection systems: A survey. *International Journal of Innovation and Applied Studies*, 5(3):233.
- Marino, L. (1990). *Il rilievo per il restauro*. Ulrico Hoepli.
- Marshall, A. R. (1989). *Network design and optimization in close range photogrammetry*. School of Surveying, University of New South Wales Kensington, N.S.W., Australia.
- Martín, A. J. and Puerto, F. P. (2003). *Levantamiento y análisis de edificios: tradición y futuro*. Colección Textos de Doctorado. Serie Arquitectura. Universidad de Sevilla.
- Martín López, F. (2015a). Cameras positioning. <https://github.com/felmarlop/cams.positioning>.



- Martín López, F. (2015b). Trabajo Fin de Grado en Tecnologías Informáticas. Aplicación Web de Posicionamiento Topográfico. Universidad de Sevilla.
- Mason, S. (1994). *Expert System-Based Design of Photogrammetric Networks*. PhD thesis, Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland.
- Ménanteau, L. and Pourinet, L. (2002). Geolittomer letg-umr 6554-cnrs.
- Ménanteau, L. and Pourinet, L. (2005). Geolittomer letg-umr 6554-cnrs.
- Mendez, N. D. D., Porras, J. C. C., and Laverde, R. M. (2007). Seguridad inteligente. *Scientia et technica*, 3(35):389–394.
- Michalewicz, Z. and Janikow, C. Z. (1991). *Handling Constraints in Genetic Algorithms*.
- Molina Font, J. (2001). *Molinos de marea de la Bahía de Cádiz : (siglos XVI-XIX)*. Junta Rectora del Parque Natural Bahía de Cádiz: Mancomunidad de Municipios de la Bahía de Cádiz, Instituto de Medio Ambiente, D.L. 2001.
- Morales, Ó. (2006). *Vigilando Poliedros Ortogonales en 3D*. PhD thesis, Universidad Nacional Autónoma de México.
- Moujahid, A., Inza, I., and Larranaga, P. (2008). Tema 2. algoritmos genéticos. *Departamento de Ciencia de la Computación e Inteligencia Artificial Universidad del País Vasco-Euskal Herriko Unibertsitatea*, pages 1–33.
- Nobile, A. (2012). *I Sistemi a Scansione 3D per la Documentazione Metrica e lo Studio Diagnostico dei Beni Culturali*. PhD thesis, Università degli Studi di Firenze.
- Ntafos, S. (1992). Watchman routes under limited visibility. *Computational Geometry*, 1(3):149–170.
- Olague, G. (2001). *Autonomous Photogrammetric Network Design Using Genetic Algorithms*, pages 353–363. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Olague, G. and Dunn, E. (2007). Development of a practical photogrammetric network design using evolutionary computing. *The Photogrammetric Record*, 22(117):22–38.
- Olague, G. et al. (2000). Design and simulation of photogrammetric networks using genetic algorithms. In *Proceedings of the 2000 Meeting of the American Society for Photogrammetry and Remote Sensing (ASPRS 2000)*, volume. Citeseer.

- Olague, G. and Mohr, R. (1998). Optimal camera placement to obtain accurate 3d point positions. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, volume 1, pages 8–10. IEEE.
- Olivito, R., Taccola, E., and Albertini, N. (2015). A hand-free solution for the interaction in an immersive virtual environment: the case of the Agora of Segesta. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:31–36.
- Ornia Núñez, F. J., Ramírez Pérez, M. P., Scroccarello, S., and Cabrera Revuelta, E. (2017). Trabajo final asignatura Nuevas Tecnologías aplicadas a los Levantamientos Arquitectónicos. Pabellón de Marruecos de la Exposición Iberoamericana de 1929.
- O’Rourke, J. (1983). Galleries need fewer mobile guards: a variation on Chvátal’s theorem. *Geometriae Dedicata*, 14(3):273–283.
- O’Rourke, J. (1987). *Art Gallery Theorems and Algorithms. The International Series of Monographs on Computer Science*. Oxford University Press, New York, NY.
- O’Rourke, J. (1998). *Computational geometry in C*. Cambridge University Press.
- Osman, I. H. and Kelly, J. P. (1996). *META-HEURISTICS: Theory & Applications*.
- Páez Parra, D. and Vargas Méndez, A. C. (2013). Trabajo fin de grado. evaluación de la eficiencia de algoritmos genéticos y sistemas multiagentes para el problema de programación de la producción en una máquina (1/prmp/?wtj).
- Parian, J. A. and Gruen, A. (2005). Close range photogrammetric network design for panoramic cameras by heuristic simulation. I:237–244.
- QGIS, E. (2017). Qgis. <http://www.qgis.org/es/site/>.
- Ramírez, J. (2008). Inteligencia artificial por Jéssica Ramírez. <http://jessy-inteligenciartificial.blogspot.com.es/2008/07/algoritmos-genticos-y-sus-aplicaciones.html>.
- Ramírez Pérez, M. P. and Cabrera Revuelta, E. (2016). Práctica asignatura Nuevas Tecnologías aplicadas a los Levantamientos Arquitectónicos. Nave lateral Ayuntamiento de Sevilla.
- Rao, S. S. (2009). *Engineering optimization: theory and practice*. John Wiley Sons.

- Real Academia Española, R. A. E. (2014). *Diccionario de la lengua española*, 23 Edición. Espasa, Madrid.
- Ríos, A. S. (2000). *Fundamentos teóricos de los métodos topográficos*. Bellisco.
- Roberts, D. and Marshall, A. D. (1998). Viewpoint selection for complete surface coverage of three dimensional objects. In *BMVC*, pages 1–11.
- Rodríguez-Gonzálvez, P., Nocerino, E., Menna, F., Minto, S., and Remondino, F. (2015). 3d surveying & modeling of underground passages in WWI fortifications. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:17–24.
- Rosser, J. F., Leibovici, D., and Jackson, M. (2017). Rapid flood inundation mapping using social media, remote sensing and topographic data. *Natural Hazards*, pages 1–18.
- Russo, M., Remondino, F., and Guidi, G. (2011). Principali tecniche e strumenti per il rilievo tridimensionale in ambito archeologico. *Archeologia e calcolatori*, 22:169–198.
- Sancho Caparrini, F. (2009). Fernando Sancho Caparrini. <http://www.cs.us.es/~fsancho>.
- Sancho Caparrini, F. (2015). Pso: Optimización por enjambres de partículas. <http://www.cs.us.es/~fsancho/?e=70>.
- Schaffer, J. D., Caruana, R. A., Eshelman, L. J., and Das, R. (1989). *A study of control parameters affecting online performance of genetic algorithms for function optimization*, pages 51–60.
- Sivanandam, S. and Deepa, S. (2007). *Introduction to genetic algorithms*. Springer Science & Business Media.
- Slama, C. C., Theurer, C., Henriksen, S. W., et al. (1980). *Manual of photogrammetry*. Number Ed. 4. American Society of Photogrammetry.
- Suarez, O. (2013). Una aproximación a la heurística y metaheurísticas. *INGE@UAN-Tendencias en la Ingeniería*, 1(2).
- Suárez Japón, J. M. (1989). *La Casa Salinera de la Bahía de Cádiz*. Fundación Machado, Consejería de Obras Públicas y Transportes y Diputación Provincial de Cádiz.

- Tang, Z.-Z., Liang, J., Guo, C., and Wang, Y.-X. (2012). Photogrammetry-based two-dimensional digital image correlation with nonperpendicular camera alignment. *Optical Engineering*, 51(2):023602–1.
- Trocadero, P. (2011). Molino de marea. <https://www.youtube.com/watch?v=GKo2tfebRgo>.
- Tucci, G., Bonora, V., Conti, A., Fiorini, L., Korumaz, A., Montanari, F., and Ronfetto (2013). Fotogrammetria stereoscopia. [http://www.geomaticaeconservazione.it/archivi/Fotogrammetria \\_Stereoscopia.pdf](http://www.geomaticaeconservazione.it/archivi/Fotogrammetria_Stereoscopia.pdf).
- Urrutia, J. et al. (2000). Art gallery and illumination problems. *Handbook of computational geometry*, 1(1):973–1027.
- Vázquez, P.-P., Feixas, M., Sbert, M., and Heidrich, W. (2001). Viewpoint selection using viewpoint entropy. In *VMV*, volume 1, pages 273–280.
- Wilensky, U. (1999). Netlogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University. <http://ccl.northwestern.edu/netlogo/>.
- Yabe, M., Goins, E., Jackson, C., Halbstein, D., Foster, S., and Bazely, S. (2015). Restoring fort frontenac in 3d: Effective usage of 3d technology for heritage visualization. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*.
- Yang, G., Dong, R., Wu, H., and Liu, C. (2014). Viewpoint optimization using genetic algorithm for flying robot inspection of electricity transmission tower equipment. *Chinese Journal of Electronics*, 23(2):426–431.